Available online at www.sciencedirect.com
ScienceDirect
Computer Networks 52 (2008) 1647–1674
Computer Networks
www.elsevier.com/locate/comnet

Design and evaluation of flow handoff signalling for multihomed mobile nodes in wireless overlay networks

Qi Wang *, Robert Atkinson, John Dunlop

Mobile Communications Group, Department of Electronic and Electrical Engineering, University of Strathclyde, Glasgow G1 1XW, UK

Received 18 May 2007; received in revised form 19 February 2008; accepted 19 February 2008
Available online 29 February 2008

Responsible Editor: W. Kellerer

Abstract

With the increasing deployment of wireless overlay networks, a mobile node with a range of network interfaces can be connected to multiple heterogeneous or homogeneous access networks simultaneously. Such host multihoming technology can be exploited to distribute (or hand off) application flows among the most appropriate interfaces and access networks dynamically to achieve end-to-end seamless, robust and even quality-of-service-aware communications for mobile users. It is essential that an efficient and effective flow handoff signalling scheme be in place. Nevertheless, little prior work has addressed this problem sufficiently in a systematic way and little performance evaluation is readily available. We propose a set of signalling procedures for a comprehensive, flexible yet standard-oriented flow handoff solution. Two candidate schemes are designed by extending and optimising related IETF work based on Mobile IPv6 or Network Mobility (NEMO). Theoretical analyses are performed and numerical results are then presented with a focus on signalling loads to compare the two proposals and to demonstrate that the designs can largely meet the requirements on desired signalling performance. Preliminary implementations and experimental results are also reported to validate the concepts of the designs, investigate the flow handoff signalling delays and verify the effectiveness of the policy-based flow handoff support for typical real-time and non-real-time applications.

Keywords: Multihoming; Flow handoff; Mobile IPv6; NEMO; Signalling performance

1. Introduction

The next-generation wireless systems are evolving towards all-IP wireless overlay networks, comprising heterogeneous or homogeneous networks whose coverage areas are overlapped [1]. A multihomed mobile node (MN) is either a single mobile host or a mobile router, together with a whole mobile network, that is equipped with multiple network interfaces. A multihomed MN has the potential to fully utilise value-added benefits from wireless overlay networks where Wi-Fi, WiMAX, and cellular systems are complementary to each other in many aspects such as coverage ranges and data rates.

* Corresponding author. Tel.: +44 141 5706041; fax: +44 141 552 4968.
E-mail addresses: qwang@eee.strath.ac.uk (Q. Wang), ratkinson@eee.strath.ac.uk (R. Atkinson), j.dunlop@eee.strath. ac.uk (J. Dunlop).

1389-1286/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.comnet.2008.02.005

In this context, it is desired and practical that both a mobile user and the network can strategically trigger a handoff to switch all or selected application flows from one interface or access network to another. We refer to such an operation as a flow handoff. Clearly, effective flow handoff support is a key enabler to achieve the "Always Best Connected" (ABC) vision [2] when coupled with intelligent network selection algorithms. From the user's perspective, a flow handoff may be triggered by either the user's movement or quality of service (QoS) preferences even in static state. For instance, a user may select an access technology with lower tariff, when available. Certain applications running in the MN may trigger a flow handoff with the help of end host measurements such as redirection of delay-sensitive application flows to the link with lower round-trip time. From the network's perspective, a flow handoff may be initiated to improve the network's service such as load sharing and fault tolerance. For example, if the network detects that one access network is overloaded it has the option of distributing a subset of the flows through another access network. Similarly, an underutilised access network could share some of the load. Detection of such network events can be fulfilled by the network more accurately. Certainly, there are many other examples that necessitate such flow handoffs to improve QoS for a mobile user, the network operator and/or the service provider. These observations justify a hybrid flow handoff approach, where both a user and the network can trigger a flow handoff, and both user- and network-triggered handoffs can be supported in a unified architecture.

In conjunction with the IST MULTINET project [3], we aim to support multihomed nomadic workers in charge of repairing and maintaining distributed high-tech and complex machines. Typically, after roaming to the scene of work, these workers would stay there working on the static machines. With the increasing deployment of wireless overlay networks, the workers often work in the coverage of multiple access networks. During the working process at the site, these workers need to communicate with their back office regularly for remote support, transmitting real-time or non-real-time data or downloading documentations and instructions. From time to time, the multiple tasks being performed would generate multiple simultaneous application flows with different QoS requirements or user/application preferences. The core objective of MULTINET is to investigate standard-oriented solutions to comprehensive flow handoffs in wireless overlay networks to accomplish robust, seamless and QoS-aware service delivery. In this paper, we focus on the design and evaluation of flow handoff signalling procedures. The design and analysis take into account flow handoffs triggered by either movement for mobile users or by intelligent network selection for nomadic users, although the latter is concentrated on in our experiments. Despite the extensive related work on handoff management and multihoming support, much work is still needed in design and evaluation. In fact, little prior work has provided a systematic design of flow handoff signalling procedures that exploit both users' and the network's intelligence; and even less has offered an in-depth analytical or experimental evaluation or comparison of promising approaches.

The remainder of this paper is organised as follows. We survey related work on mobility and multihoming in Section 2. Section 3 presents our motivations for flow handoff support signalling schemes, the overview of the proposals and the reference network model. In Section 4, we expound our proposed signalling schemes towards a comprehensive and standard-oriented solution. A set of procedures are illustrated and described including both user- and network-triggered flow handoffs and supporting routines. The theoretical analyses and numerical results of the proposed schemes are then provided in Sections 5 and 6, respectively. Subsequently, implementations and experimental results are presented in Section 7 to complement the analytical results and validate the concepts of the proposed designs. Finally, Section 8 concludes this paper.

2. Related work

2.1. Mobile IPv6-based approach

The IETF plays an essential role in IP-based mobility management. Mobile IPv6 (MIPv6) [4] is the de facto standard for IPv6 mobility support. Unfortunately, MIPv6 in its current form does not support advanced multihoming beyond handing off all the flows from one interface to another. A number of MIPv6 variants such as Hierarchical MIPv6 (HMIPv6) [5] and Fast Handovers for MIPv6 (FMIPv6) [6] have been proposed for performance optimisation and extension. In particular, Network Mobility (NEMO) [7] extends the IP mobility support from a single mobile host (MH)

to a whole mobile network managed by a mobile router. However, MIPv6 and these variants have not addressed advanced flow handoffs in a multihoming context. Following the end-to-end principle in the Internet design, the IETF has concentrated on the user-centric approach though there is a need for the network-supported approach [8] and thus the Network-based Localised Mobility Management (NETLMM) WG has been launched.

Recently, the Mobile Nodes and Multiple Interfaces in IPv6 (MONAMI6) WG has been standardising MIPv6/NEMO-based host multihoming mechanisms to facilitate flow handoffs for multihomed MNs. Multiple Care-of-Addresses (CoAs) registrations are allowed in [9] so that a single Home Address (HoA) can be bound with more than one CoA through a pair of Binding Update (BU) and Binding Acknowledgement (BA) messages. To distinguish the different (HoA, CoA) bindings for a given MN, an extra identifier called the Binding Unique ID (BID) is introduced. Typically, each BID corresponds to and identifies a specific interface of the MN.

Furthermore, the flow bindings draft [10] enables a particular flow to be bound with a CoA associated with an interface. A Flow ID option accommodates the flow identifier such as a subset of the five-tuple (source and destination addresses and port numbers, transport protocol), e.g., the well-known port numbers can identify different application flows (e.g., 8080 for HTTP flows, 20 for FTP flows). The Flow ID option, also placed in a BU or BA message, can indicate adding, replacing or deleting of a flow binding policy (Flow ID, BID), identified by a Flow ID identifier. A default binding exists in case of no matching. A major problem in this proposal is that only user-initiated flow handoffs are addressed as a BU is always sent from a MN to its Home Agent (HA), although both user- and network-initiated flow handoffs are desired and should be supported.

The flow distribution draft [11] targets the similar task from a different approach. An XML schema fragment is defined to code the policies and applies the Simple Object Access Protocol (SOAP) [12] over HTTP or HTTPS web service to deliver the messaging. Since SOAP request and reply messages can be send from a MN or its HA, potentially network-triggered flow handoffs could be introduced. Nevertheless, a dedicated intelligent network selection server would be more advantageous to prevent the HA from overloading. No such server or network intelligence to trigger the flow handoffs has been reported.

Furthermore, the two approaches presented in [10,11] lack a systematic design and presentation of a full set of signalling procedures needed for flexible policy-based flow handoff support. In addition, numerical comparisons based on either theoretical analysis or implementations between them are missing. We attempt to fill these gaps in this paper.

2.2. SCTP-based approach

The Stream Control Transmission Protocol (SCTP) [13] is an emerging transport protocol that supports multihoming in its own right. SCTP multihoming allows binding of one transport layer's connection to multiple IP addresses at each end of the connection. This binding allows a sender to transmit data to a multihomed receiver through different destination addresses. Simultaneous transmission to multiple destination addresses is being investigated. For instance, Iyengar et al. [14] proposed and analysed concurrent multi-path transfer (CMT) between multihomed source and destination hosts to increase an application's throughput. Note that this is different from the simultaneous use of multiple interfaces for concurrent different applications' transmissions and their dynamic flow handoffs among the interfaces in our study.

The multihoming capability in SCTP has also been employed for handoff management such as in mSCTP [15] and mobile-SCTP [16]. Those schemes utilise end-to-end semantics for signalling handoffs. The end-to-end approach would reduce the home registration delay and eliminate the tunnelling cost that exists in the Mobile IP-based handoff management protocols. On the other hand, the end-to-end paradigm lacks the support from network intelligence that can be very beneficial for overall QoS provision. In our design, network intelligence is exploited to trigger network-initiated flow handoffs.

After all, the SCTP multihoming is mainly designed to enable retransmissions to alternate IP addresses for survivability when the primary IP address becomes unavailable. There is little, if any, SCTP-based work aims to enable policy-based handoffs of different application flows among the interfaces of a multihomed node. More importantly, SCTP multihoming is operating in the transport layer and thus it would only benefit applications that are based on SCTP rather than TCP or UDP, over which most IP applications are currently running. Therefore, although SCTP-based multihoming handoff management is a promising approach, we

believe that a network-layer solution e.g., enhanced MIPv6/NEMO would be more appropriate in the near future.

2.3. Other relevant work

The Host Identity Protocol (HIP) [17] is another promising proposal that could facilitate IP mobility and multihoming. HIP introduces a new "host" layer between the network and the transport layers. Consequently, flows are bound to host identities instead of IP addresses so that the change of IP addresses can be transparent to the applications as in the basic MIPv6/NEMO. Regarding multihoming, similar to SCTP alternate IP addresses can be exchanged between the MN and its peer so that survivability can be achieved. In addition to the similar limitation as found in SCTP-style multihoming, adding a new layer to the protocol stack is not a minor modification and this may cause an updating of numerous IP applications. Hence, HIP-based multihoming may not be preferred in the short to middle term either.

It is also noted that work is underway on site multihoming, where a site's network has connections to multiple IP service providers. The IETF Site Multihoming in IPv6 (MULTI6) WG defines a base architecture and the Site Multihoming by IPv6 Intermediation (SHIM6) WG is developing the protocols. Our current research concentrates on host multihoming although advances in site multihoming may be complementary in improving performances.

The IETF Detecting Network Attachment (DNA) WG is extending the current IPv6 host auto-configuration protocols to allow MNs to detect their IP layer configuration and connectivity status more quickly so that the current MIPv6/NEMO handoff performance would be optimised. The IEEE 802.21 [18] is defining new signalling procedures and APIs related to the link layer for L2 triggers and network selection. These schemes intend to complement the IETF IP mobility protocols. Our proposals follow the IETF track.

In addition to the standardisation work, there is an increasing interest in multihoming in the wider research community. TCP traffic multihoming is supported in [19] with the IP Network Address Translator (NAT) function. IP-in-IP tunnelling is employed in [20] to aggregate the bandwidth of multiple IP paths for improved throughput. The Router Advertisement messages are extended in [21] to enforce multihoming signalling between a MN and an access router (AR) for load balancing and fault tolerance. These studies did not address advanced policy-based flow handoffs based on MIPv6/NEMO. Regarding handoff decision-making algorithms for multihomed MNs, Wang et al. [22] proposed a policy-enabled scheme with a cost function covering available bandwidth, power consumption and service tariff in heterogeneous networks. In [23], the handover decision-making process uses context information regarding user devices and location, network environment and requested QoS. More algorithms are designed in [24] to optimise the costs and performances for multihomed users. The policy defined in [22] and the pilot algorithms proposed in [23,24] may serve as a subset of the desired intelligent network selection algorithms, which are beyond this paper's scope. In addition, host multihoming would also facilitate seamless handoffs from another approach compared with FMIPv6 as reported in [25]. The work in [25] complements our focus of this paper on handoffs triggered by network selection intelligence other than user movement.

3. Proposed system overview

3.1. Motivations

The work-in-progress in the IETF MONAMI6 WG has laid a foundation for flow handoff support. In our design, these separate proposals are exploited as building blocks towards a comprehensive unified architecture. Refs. [10,11] provide a good start towards flexible policy identification and distribution for flow handoffs. However, neither a detailed analysis of such solutions nor a comparative study with each other is available. Furthermore, to support advanced flow handoff signalling, a complete set of procedures and some key components are still missing in both approaches as mentioned in Section 2.1.

Our contributions in this paper are threefold. Firstly, we attempt to design two complete sets of flow handoff signalling schemes by enhancing and extending the approaches in [10,11], respectively. Secondly, we present an original and pioneering numerical comparison of both approaches in terms of signalling efficiency through an analytical methodology. Thirdly, we validate our designs with preliminary implementations of both approaches, and further evaluate and compare their performances in terms of flow handoff signalling delays. The effectiveness of the flow handoffs are also

verified for both real-time and non-real-time applications.

Compared with the existing work, we take a more systematic design approach to flow handoffs for multihomed MNs with a set of design requirements considered. Functionally, the system should support both user- and network-triggered flow handoffs in a unified platform to satisfy the practical demands from both sides' perspectives. This would allow flexible generation of handoff triggers wherever appropriate and convenient, as aforementioned; and fully utilise the intelligence of both the MN and the network. The signalling system should coordinate the MN and the network and fulfil the synchronisation of the flow binding policies at both sides. Moreover, it may be desirable that a user be granted a negotiation option in case of network-triggered handoffs. Architecturally, the system should facilitate an incremental deployment. Thus, a centralised paradigm may be established in the first stage since such a paradigm is more easily manageable. In a later stage, a more distributed paradigm may be deployed for optimised performances.

Regarding the performance of the proposed signalling schemes, signalling efficiency is among the top concerns of any signalling system. Efficient signalling is required for bandwidth-limited wireless networks. Decent handoff signalling delays due to policy distribution and enforcement are also desired so that a policy-based flow handoff can take effect quickly and smoothly. As to implementations, it is desirable that the proposed system should raise as little concern as possible in implementation convenience and operation interoperability. Standard-oriented schemes should alleviate such concerns. In addition, good message readability and extensibility are preferable to the signalling designers and developers.

3.2. Overview of the proposals

We have designed two independent standard-oriented signalling schemes. Briefly, Scheme I extends the flow bindings draft [10] by enabling flow binding signalling from the HA to a MN for network-triggered flow handoffs. Following the tradition in MIPv6, we define new ICMPv6/UDP messages for supporting signalling such as network triggers and probes. Alternatively, Scheme II enhances the flow distribution draft [11] by defining new SOAP messages for the supporting signalling. Both schemes reuse the multiple CoA registration procedure [9] built upon MIPv6/NEMO. Therefore, both schemes are based on standard protocols and the interoperability should not be of great concern.

Furthermore, both schemes are designed to meet a set of challenges. They support both kinds of triggers and they can fall back to user triggers only if the required network selection algorithm is unavailable. Detailed intelligent network selection algorithms are beyond the scope of this paper as we concentrate on the signalling aspects. Support of gradual deployment is also taken into account. In the current stage, the HA is enhanced to be in charge of handling multiple CoA registrations of a MN, synchronising flow binding policies with the MN, and handing off selected downlink flows according to the up-to-date policies. In our future work, the Mobility Anchor Point (MAP) defined in HMIPv6 would be enhanced to act as a virtual local HA in a foreign domain for the visiting MNs so that most of the signalling to the HA can be localised and the workload at the HA can be distributed.

For secure signalling, MIPv6/NEMO has recommended IPsec [26,27]. The same security mechanism is also applicable to the proposed schemes, which are IP-based protocols. Therefore, standard secure signalling can be achieved in our design without inventing new specific security mechanisms. Further discussions on Authentication, Authorisation and Accounting (AAA) are beyond the scope of this paper.

Both schemes can operate over the reliable TCP or the unreliable UDP. When UDP is used, a signalling protocol (MIPv6/NEMO, ICMPv6 or SOAP) needs to employ its own timer-based retransmission mechanism in the same manner as defined in the MIPv6 specification [4]. Signalling reliability can thus be provisioned to ensure successful end-to-end message delivery. Secure and reliable signalling is further considered and explained in Section 5 to derive the signalling loads.

Finally, the messages defined in both schemes are extensible. In particular, the textual XML-coded SOAP messages appear more readable and modifiable to developers, especially in a joint project like MULTINET.

3.3. Reference model

The network model is illustrated in Fig. 1 to facilitate the description of the subsequent design and analysis. It is a simplified version of the MULTINET

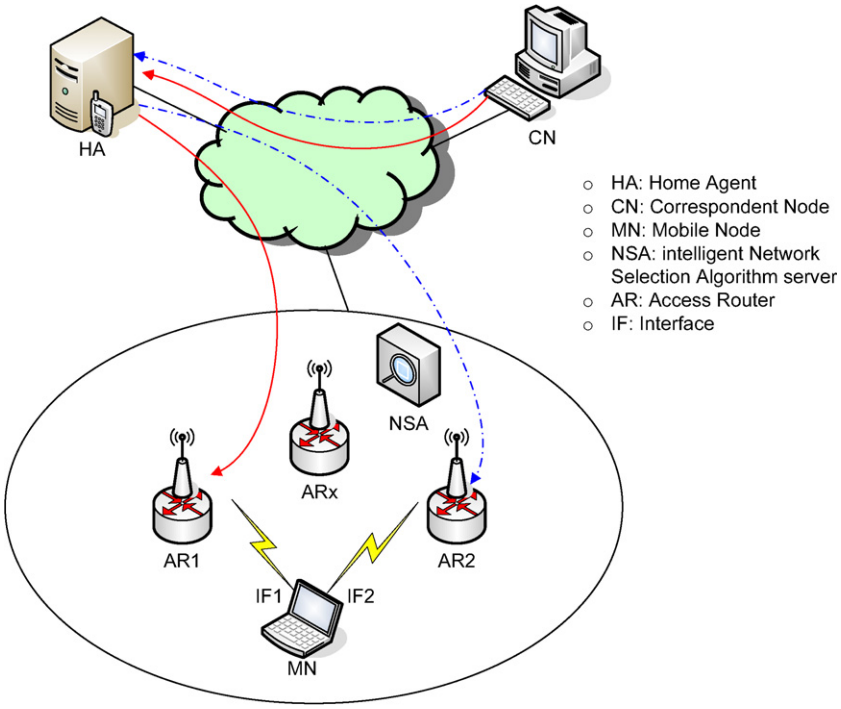


Fig. 1. Reference network model.

reference architecture [3]. In this model, a MN can be equipped with multiple network interfaces though two interfaces IF1 and IF2 are demonstrated. The MN under consideration is visiting a foreign domain. The HA is located in the home domain and it collaborates with the MN and the intelligent Network Selection Algorithm server (NSA) for flow handoff support. Symmetric flow binding policies are stored, synchronised and enforced in both the MN and the HA for distributing the identified uplink and downlink flows, respectively. A correspondent node (CN) is stationary in a third domain. All the domains are interconnected to each other through a common IP core network. The MIPv6 bi-directional tunnelling mode between the MN and the HA is demonstrated for incremental deployment and NEMO base protocol compatibility. Fig. 1 only shows the downlink flows for presentation clarity. An additional advantage of this mode is that the CN can be unaware of the MN's location or being multihomed. The CN always sends packets to the MN's HoA. The route optimisation or HMIPv6 adoption would be investigated in future work.

In the foreign domain, two access routers AR1 and AR2 provide wireless access to the MN initially. Subsequently, the MN may be connected to other ARs, e.g., ARx, on the user's movement. Nevertheless, it is important to notice that the MN can experience flow handoffs due to either MN movement or intelligent network selection for dynamic MN/network QoS-driven adaptation. A serving NSA collects and processes periodic network QoS measurement reports from the ARs, and determines if the network should initiate a network-triggered flow handoff based on the output of the intelligent network selection algorithms running in the NSA. Note that there is no constraint on the physical location of the NSA: it can be located wherever appropriate as chosen by the service provider. For instance, more than one foreign domain may share a NSA in a gradual deployment. Security associations between the signalling entities are established for secure signalling. Pure IPv6 is assumed in this model although the support for IPv4–IPv6 transition is also considered [3].

4. Proposed signalling design

In this section, we describe the proposed signalling procedures in the two schemes aforementioned. We focus on the signalling protocols with involved messages briefed. The detailed message formats are defined whereas they are not expounded here for conciseness.

4.1. Initial registration of multiple CoAs

To obtain the flow handoff support for downlink traffic and network-triggered flow handoff notification, a MN must configure and register the multiple CoAs assigned to its interfaces at the HA as the first step. This kick-off procedure is illustrated in Fig. 2.

On being powered up in a foreign domain, the MN may wait for the next scheduled unsolicited Router Advertisement (RA). Since an unsolicited RA may not arrive promptly or may not include the complete prefix information [28], the MN may alternatively send a Router Solicitation (RS) from each of its interfaces to request a RA from the corresponding ARs. A RS is typically sent to the All-Routers multicast address [28]. Consequently, all of its interfaces can be configured with a unique IPv6 address (CoA) through either the default IPv6 stateless address auto-configuration [29] or an optimised variant that can accelerate the process and reduce the overheads. We assume the latter approach as shown in Fig. 2. The signalling for this procedure is the same in Scheme I and Scheme II. "{...}" means a specific reply to the corresponding request. Once the interfaces are configured with unique IPv6 addresses, the MN sends one or more BU messages to the HA to register these CoAs as defined in [9]. To reduce the signalling loads, the bulk registration is preferred so that the multiple CoA registration is performed by a single BU. After verifying and processing the BU, the HA then replies with a BA indicating the success or failure (with the corresponding error code) of the registration.

4.2. Default flow binding policy and profile registration

Subsequently, as shown in Fig. 3 the MN should register its default flow binding policies and user

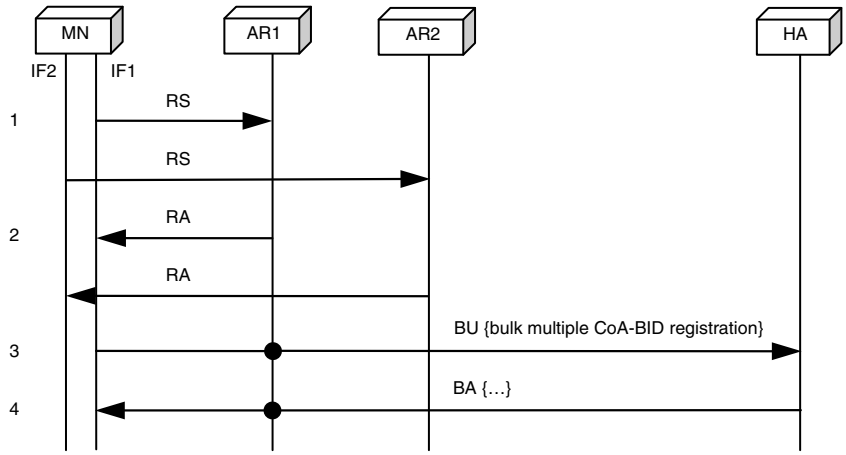


Fig. 2. Multiple CoA registration in Schemes I and II.

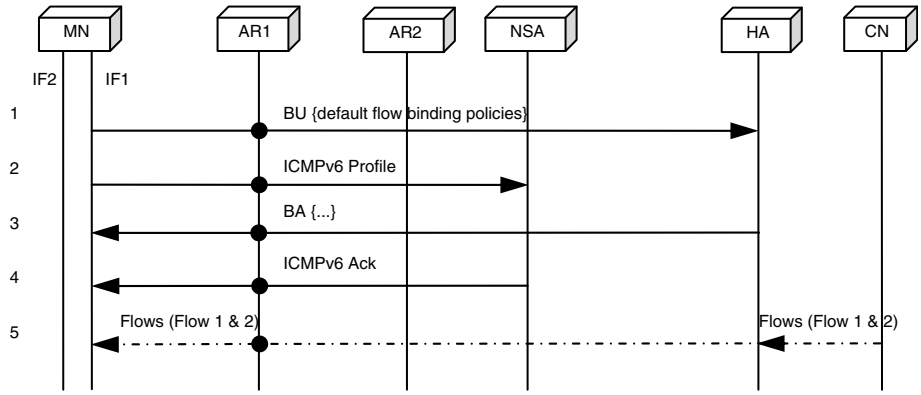


Fig. 3. Default user flow binding policy and profile registration in Scheme I.

profile with the HA and the NSA, respectively. In Scheme I, the MN initiates these tasks by sending a BU and an ICMPv6 Profile message to the HA and the NSA, respectively. The HA and the NSA then reply with a BA and an ICMPv6 Ack message accordingly. The BU requests to register the default flow binding policies, which are a set of (Flow ID, BID) bindings. Each Flow ID is a subset of the five-tuple aforementioned. The BU also indicates the default flow binding priority for each BID. When a flow does not match any flow binding policies, it will be bound to the BID that has the highest default flow binding priority. The Profile message specifies the user information, terminal information, and application preferences regarding network conditions. Note that it is possible to combine this procedure with the initial multiple CoA registration in Scheme I. The two procedures are designed to be distinct since some users may only be interested in registering multiple CoAs.

In Scheme II, pure SOAP messages are used to fulfil the above signalling. A pair of SOAP Policy or Profile Registration request and reply messages are exchanged between the MN and the HA or between the MN and the NSA, respectively. To facilitate the processing of specific policies and reduce policy refresh loads, a Policy ID similar to the FID identifier in the Flow ID option defined in [10] is introduced to optimise [11].

4.3. User-triggered flow handoffs

On a user-triggered flow handoff, the MN would update specific policies including modifying, deleting or adding selected policies at the HA. This procedure manages two scenarios depending on whether a user handoff is triggered by the user's movement or not. The signalling varies according to the scenarios or the schemes.

In Scenario A, one (or more) of the interfaces of the MN connects to a new AR due to movement. After the movement is detected through a link-layer trigger (e.g. [18]) or a network-layer mechanism (e.g. upon receiving an unsolicited RA from the new AR), the MN sends an AS to the All-Routers multicast address or the new AR directly if possible, and the new AR in turn replies with a RA as shown in Steps 2A-1 and 2A-2. Consequently, the MN obtains a unique IPv6 address for that interface through the IPv6 stateless address auto-configuration or an optimised variant. Here we assume that an optimised router detection and address distribution scheme (e.g., [18,30]) is in place to minimise the otherwise considerable overheads and delay in the movement detection [31] and the duplicate address detection processes. In Scheme II, a pair of BU and BA is exchanged for new CoA registration in Steps 2A-3 and 2A-4; and if flow binding policy update is needed, a pair of SOAP messages are used in Steps 2A-5 and 2A-6. In contrast, in Scheme I, both new CoA registration and the possible policy update can be performed through one pair of BU and BA in Steps 2A-3 and 2A-4. Note that if the new AR uses a homogeneous radio technology the MN would typically only need to update the CoA(s) associated with the involved BID(s) and leave the policies intact since the policies are bindings between BIDs and Flow IDs.

In Scenario B, the MN, even in a stationary state, triggers a flow handoff based on its own intelligence for better connections. As an example, we assume that two flows (flow 1 and flow 2) have been established between the CN and the MN as shown in Step 1. When a flow handoff is user-triggered, the MN sends a BU with the new flow binding policies enclosed to the HA, as depicted in Step 2B-1. Note that the MN may prefer to use the targeted (secondary) interface for the signalling especially when the source interface is going down. We assume that the new flow policies indicate that one of the flows (flow 2) needs to be handed over from the current interface to the secondary one. Upon receiving the BU, the HA authenticates and verifies the BU. If the authentication and verification are successful, the HA updates the preinstalled flow binding policies of the MN and replies with a BA in Step 2B-2. Afterwards, the HA hands off flow 2 to its interface connected to the access network corresponding to the MN's secondary interface. The handoff is achieved by tunnelling (IP-in-IP encapsulation). Consequently, flows are distributed between the interfaces as desired in Step 3.

In the illustrations, we assume that the MN uses the new CoA in Scenario A or the existing CoA associated with the targeted interface in Scenario B for CoA registrations and/or flow binding policy updates. In addition, the illustrations only demonstrate the downlink flows, and selected downlink flows are handed over to another access network by the HA on a flow handoff. Regarding an uplink flow handoff, the MN itself switches selected flows to another interface by tunnelling in a similar manner.

4.4. Network-triggered flow handoffs

On a network-triggered flow handoff, the HA updates specific policies at the MN based on a trigger from the NSA. Fig. 5 depicts this procedure in Scheme I. As shown in Step 2, the monitor module collocated with a AR measures the QoS metrics and sends the measurements in a Probe message to the NSA periodically. A Probe message provides the current QoS of the access network in terms of a series of selected metrics. Four metrics have been defined including traffic load, available bandwidth, Received Signal Strength Indication (RSSI) and Signal-to-Noise Ratio (SNR). The NSA collects and input these QoS measurements to the predefined intelligent network selection algorithms. Assuming a flow handoff is to be triggered as shown in Step 3, the NSA sends a Trigger message to the HA in Step 4. A Trigger message comprises one or more triggers, each of which is a binding of a Flow ID and a AR prefix. More than one Flow IDs can be bound with a same AR prefix. A Flow ID used in a trigger is typically a subset of a three-tuple including the source and destination port numbers and the transport protocol since the source and destination addresses are typically unavailable at the NSA. Triggers for multiple MNs can be enclosed in a single Trigger message for scalability. MNs with similar user profiles can be classified into the same service class and may be triggered under the same network conditions. The HA would authenticate and verify the request and acknowledges it in Step 5.

Subsequently, a notification process with a negotiation option between the network (the HA triggered by the NSA) and the user (the MN) is proposed as follows. Firstly, the HA formulates the new flow binding policies based on the received trigger. This involves a mapping operation between

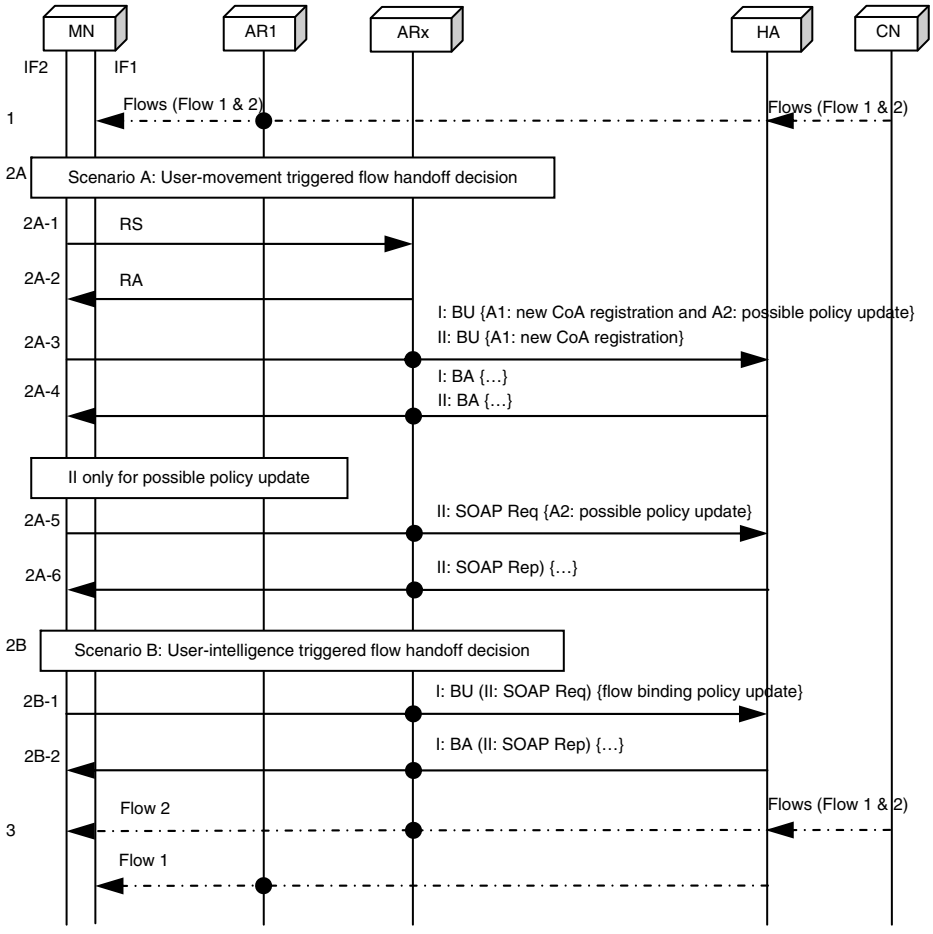


Fig. 4. User-initiated flow handoff in Schemes I and II.

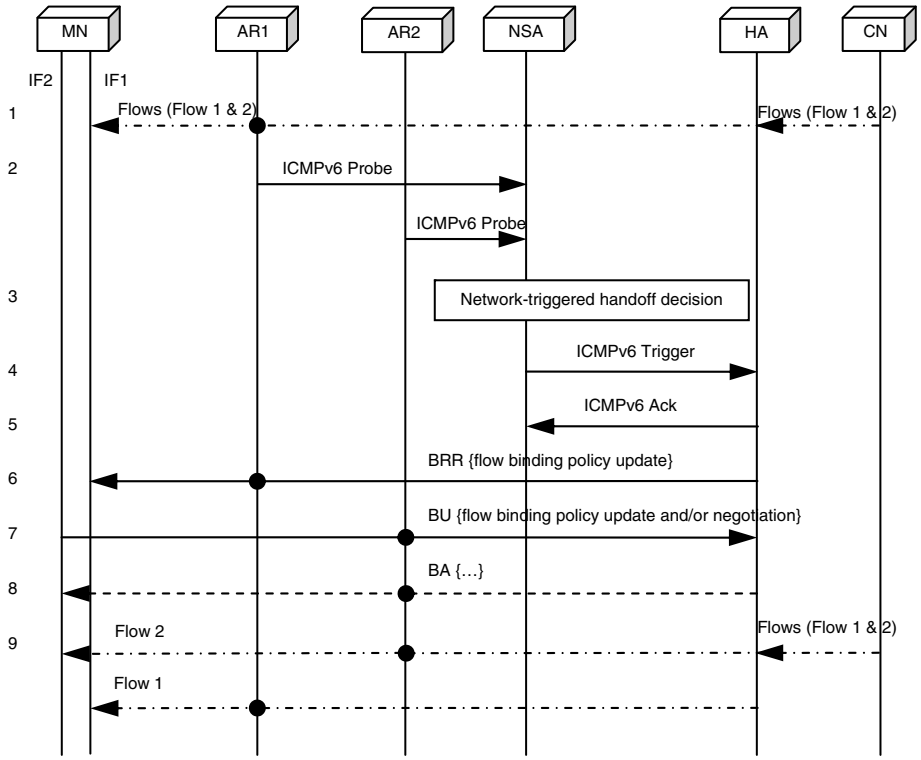


Fig. 5. Network-initiated flow handoff in Scheme I.

the ARs' prefixes and the BIDs of the involved MNs by matching the AR's prefixes and the corresponding CoAs' prefixes. Afterwards, the changed policies need to be signalled from the HA to the MN. This can be accomplished by extending the MIPv6 Binding Refresh Request (BRR) message by introducing a new mobility option for the modified policies, and allowing the HA to send such an extended BRR to the MN (Step 6). The mobility option is based on the Flow ID option defined in [10]. An optional new flag could also be defined in the option to indicate if the new policies are negotiable. Note that such an extension is supported by the extensibility of the BRR message as defined in the MIPv6 specification [4]. By extending the BRR message rather than defining a brand new mobility message, we can largely reuse the BU message (and optionally the BA) enhanced for user-initiated handoffs thanks to the existing MIPv6 BRR-BU signalling logic as to be shown in the subsequent steps. Moreover, the built-in security specified in the original MIPv6 messages can be naturally utilised.

Secondly, on receiving the BRR, the MN may accept or reject the new flow binding policies by sending back a BU with an explicit reply, e.g. by repeating the new policies if accepted or repeating the existing policies if rejected, or simply by setting a flag. This is performed in Step 7. Optionally, the HA may acknowledges with a BA in Step 8. In this BA, the HA may confirm the final decision. It is noted that either the MN's current interface in use (IF1) or the secondary interface (IF2) can be used for the above signalling since the CoAs associated with these interfaces have already been registered at the HA and they are refreshed regularly to maintain their validity. The use of the interfaces in Fig. 5 is for demonstration only though such a use may be preferred for the HA to double-check their availability. Consequently, as an example, one of the two flows is redirected to another interface by the HA tunnelling as shown in Step 9.

Scheme II uses all SOAP messages for the probes, triggers, and policy update. In addition, it may worth mentioning that it would be also possible for the HA to advise policies in user-initiated flow handoffs or even for the default flow handoff policies. In that case, the HA should include an extended Binding Refresh Advice option (the base option is defined in MIPv6 [4]) in the BA message when replying the MN's BU message.

4.5. Flow binding policies and multiple CoA registrations refresh

To keep alive the multiple CoA registration and the flow binding policy registration at the HA, the MN needs to periodically refresh these registrations before they expire. Fig. 6 illustrates this procedure in both schemes.

In Scheme I, the MN sends a BU to the HA according to the predefined lifetime of the registrations. Since the CoAs and the policies share the same lifetime, this single BU is used to refresh both multiple CoA registration and flow binding policies simultaneously. The MN replies with a BA to the MN. In the BU and the BA, for each policy only the first eight-bytes including the FID identifier other than the whole Flow ID option are enclosed. Scheme II has to use an extra pair of SOAP Registration refresh request and reply messages with the Policy ID identifiers enclosed in addition to a pair of BU and BA message for policy refresh and CoA refresh, respectively.

4.6. Deregistration

Finally, to deregister the flow handoff service the MN would initiate the deregistration procedure as shown in Fig. 7. The MN may deregister selected CoAs by sending a BU to the HA whilst retaining at least one CoA if it is still in a foreign domain and would like to receive standard MIPv6 service. Usually a MN deregisters all of the CoAs only when returning the home domain. The MN may send another BU or a SOAP message to the HA to

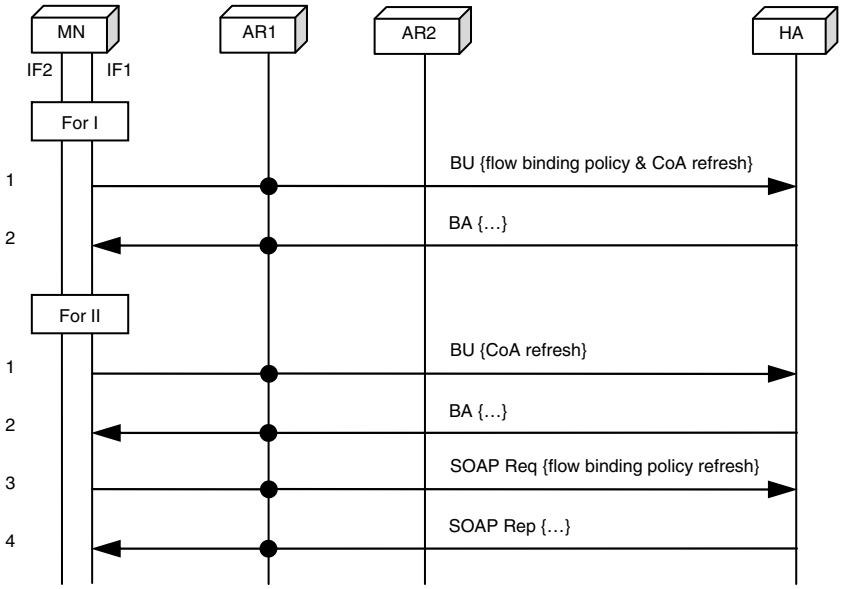


Fig. 6. Registrations refresh in Schemes I and II.

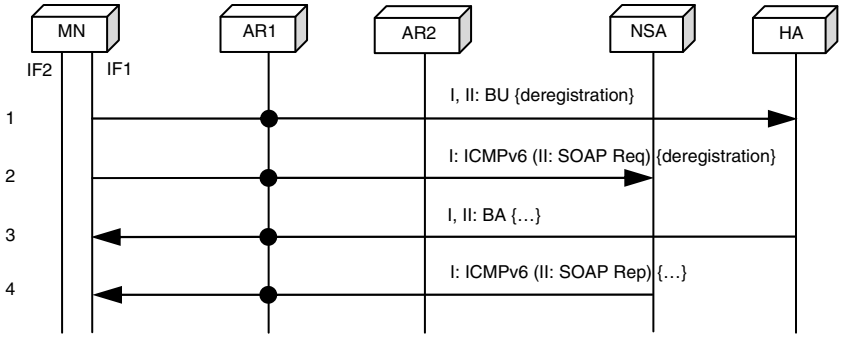


Fig. 7. Deregistration in Schemes I and II.

explicitly rearrange the flow binding policies associated with the deregistered BIDs by defining new flows and flow bindings or just rebinding existing Flow IDs to the desired BIDs. Otherwise, by default the flows bound to these deleted BIDs are then bound to the BID with the highest default flow binding priority set in the default user flow binding policy and profile registration procedure. For simplicity, we assume the latter case. Meanwhile, the MN should send an ICMPv6 or SOAP Deregistration request message to the NSA should it want to deregister the network-triggered flow handoff support service.

5. Signalling analysis

In this section, we develop an analytical model for signalling performance evaluation of the two proposed schemes.

5.1. Evaluation metric and associated parameters

In the evaluation of a mobility protocol, the major concern would be the signalling loads [32–34]. Plethoric signalling loads tend to over-consume the valuable bandwidth of the wireless network, and the processing capacity of the involved servers, and thus may lead to system performance degradation and affect the committed QoS of the services. The contribution of an individual message to the network load depends on the message length and the sequence of visited network nodes on the path between its origin and destination [32]. Therefore, the signalling loads generated by a message can be calculated as the product of the message length and the number of hops it traverses between the source and the destination in the network [33]. The aggregate signalling loads generated by a protocol are the summation of the loads contributed by all the involved individual messages. In this paper, the average (or mean) signalling loads per MN are defined as the expected accumulative IP-level messaging traffic for a specific signalling procedure or all kinds of signalling procedures related to flow handoffs in a unit time (per hour). In the following, we list the main assumptions and parameters for the subsequent analyses.

Assumptions:

Firstly, the involved entities in the signalling system are all interconnected through wired links except that a MN and the ARs are communicating through wireless links. The wired hops in the system are assumed error-free and broadband. The packet loss is only caused by the error-prone wireless hop between a MN and a serving AR. In addition, the number of hops between two entities is the same for both downlink and uplink signalling.

Secondly, SOAP run over UDP directly instead of over HTTP and the underlying TCP. In the wireless networking context, SOAP over UDP directly would be significantly more efficient [35,36]. It is reported in [36] that SOAP over UDP can reduce message size by almost 50% compared with SOAP over HTTP. Therefore, SOAP over UDP is preferred in Scheme II although both HTTP/TCP and UDP can be used. Running over UDP, both Schemes I and II apply the timer-based retransmission mechanism for recovery of lost/corrupted packets as defined in MIPv6 [4]. There are no additional retransmission (or other error correction) mechanisms. Every time when a local predefined timer expires, the sender of a request message retransmits the request. This operation is repeated until the sender receives a matching reply message from the receiver or the predefined maximum retransmission trials have been conducted. The receiver of a request sends back or retransmits a reply only upon the receipt of the correct request (either the initial one or a retransmitted copy).

Parameters:

- β_i the bit error rate (BER) of the wireless hop that packet i (a packet or a non-oversized message of type i) is being transmitted over ($\beta_i < 1$)
- α_i the packet loss rate of packet i
- $q_{i,i+1}$ the probability that a transaction consisting of a pair of request and reply packets i and $i+1$ fails (we denote packet $i+1$ be the reply to packet i that is a request)
- L_i^j the length of packet i in procedure j in the unit of bytes
- H_{A-B} the number of hops between entities A and B, or between B and A (non-directional)
- C_S^j the signalling loads incurred by procedure j in Scheme S ($S = \text{I or II}$)
- $C_{S:i}^j(h)$, $C_{S:i,i+1}^j(h)$ the signalling loads incurred by packet i in one direction or by packets i and $i+1$ in each transmission direction of h hops in procedure j of Scheme S
- λ^j the mean rate at which procedure j is invoked ($\lambda^{\text{Init-MCoA}}$, $\lambda^{\text{Def-policy}}$, $\lambda^{\text{Usr-HO}}$, $\lambda^{\text{Net-HO}}$, $\lambda^{\text{Policy-Ref}}$, and λ^{Dereg} denote the mean rate for initial multiple CoA registration, default policy and profile registration,

user-triggered flow handoff, network-triggered flow handoff, policy and CoA fresh, deregistration, respectively)

λ^{Probe} the mean frequency a Probe message is sent by a AR

p_{mv} the probability that a user-triggered flow handoff is caused by the user's movement

$p_{\text{mv-npu}}$ the probability that a user-triggered flow handoff on a user's movement does not invokes a policy update

$p_{i(j)}$, $p_{i,i+1(j)}$ the probability that a request packet i is or both packets i and $i+1$ are successfully transmitted in the jth trial, respectively

N_{probe} the average number of probe messages received at the NSA between two consecutive network-triggered flow handoffs

N_{AR} the average number of involved ARs that periodically send Probe messages to the NSA in the MN's current service area

$N_{\text{MH-tr}}$ the average number of involved MNs (MHs) in a network-triggered flow handoff

$N_{\text{HO-IF}}$ the average interfaces that need to configure a new CoA on a user-triggered flow handoff due to the user's movement

N_{RM} the maximum number of transmission trials

MTU_{min} the minimum IPv6 Maximum Transmission Unit

5.2. Signalling loads analysis

In this subsection, we derive the formulae of the signalling loads. The average signalling loads generated per MN per unit time by Scheme S consisting of m procedures are calculated as

$$C_S = \sum_{j=1}^{m} \left(\lambda^j \cdot C_S^j\right) = \sum_{j=1}^{m} \left[\lambda^j \cdot \sum_{i=1}^{l_j} \left(P_{S:i}^j \cdot C_{S:i}^j\right)\right], \quad (1)$$

where l_j is the number of messages invoked in procedure j, and $P_{S:i}^j$ is the probability that message i is invoked in procedure j. The signalling loads incurred by message i in procedure j in Scheme S are calculated as the product of its length and hops: $C_{S:i}^j = L_{S:i}^j \cdot H_{S:i}^j$[33]. For a given procedure j, the average loads are given by

$$C_S^j = \sum_{i=1}^{l_j} \left(P_{S:i}^j \cdot L_{S:i}^j \cdot H_{S:i}^j\right). \quad (2)$$

In the following, we derive the signalling loads generated by a pair of request and reply packets and then those generated by specific procedures. First of all, a successful transmission of packet i requires that every bit of the packet is correctly received. Therefore, the packet loss rate of packet i is given by

$$\alpha_i = 1 - (1 - \beta_i)^{8 \cdot L_i}, \quad (3)$$

where $8 \cdot L_i$ is the length of packet i in the unit of bits. Note that a transmission of a pair of request and reply packets would fail when either the request is lost or the request is received whereas the reply is lost. Therefore, the probability that a transmission fails and thus a retransmission is incurred is given by

$$\begin{aligned} q_{i,i+1} &= 1 - (1 - \alpha_i) \cdot (1 - \alpha_{i+1}) \\ &= \alpha_i + (1 - \alpha_i) \cdot \alpha_{i+1}. \end{aligned} \quad (4)$$

The signalling loads provoked by a pair of request and reply packets consist of two portions: the loads due to the final successful transmission and the loads from the unsuccessful transmission trials. Firstly, the mean loads generated by a successful transmission of both packets on the jth trial (after $j-1$ unsuccessful trials) over h hops including one wireless hop in each direction are given by

$$\begin{aligned} C_{i,i+1(j)\_1}(h) = q_{i,i+1}^{j-1} &\cdot (1 - q_{i,i+1}) \\ &\cdot [(L_i + L_{i+1}) \cdot h]. \end{aligned} \quad (5)$$

Next, we calculate the loads from the unsuccessful trials. When the request is sent from a MN to a network entity e, we refer it to as an uplink request; when the request is sent in the opposite direction, it is called a downlink request. The wireless hop is the first hop for an uplink packet whilst the last hop for a downlink packet. For an uplink and a downlink request respectively, the mean signalling loads generated by an unsuccessful transmission is given by

$$C_{i,i+1}^{\text{fail}}(h) = \begin{cases} \alpha_i \cdot (L_i \cdot 1) + (1 - \alpha_i) \cdot \alpha_{i+1} \\ \quad \cdot [(L_i + L_{i+1}) \cdot h] \quad \text{Uplink request;} \\ \alpha_i \cdot (L_i \cdot h) + (1 - \alpha_i) \cdot \alpha_{i+1} \\ \quad \cdot (L_i \cdot h + L_{i+1} \cdot 1) \quad \text{Downlink request.} \end{cases} \quad (6)$$

The mean accumulative loads generated by the previous $j-1$ unsuccessful trials are then given by

$$\begin{aligned} C_{i,i+1(j)\_2}(h) &= q_{i,i+1} \cdot C_{i,i+1}^{\text{fail}}(h) + q_{i,i+1}^2 \cdot C_{i,i+1}^{\text{fail}}(h) \\ &\quad + q_{i,i+1}^3 \cdot C_{i,i+1}^{\text{fail}}(h) + \cdots + q_{i,i+1}^{j-1} \cdot C_{i,i+1}^{\text{fail}}(h) \\ &= C_{i,i+1}^{\text{fail}}(h) \cdot q_{i,i+1} \frac{1 - q_{i,i+1}^{j-1}}{1 - q_{i,i+1}}, \quad q_{i,i+1} < 1, \end{aligned} \quad (7)$$

where $h = H_{AR\text{-}e} + 1$, and $H_{AR\text{-}e}$ denotes the hops between the AR and the network entity e. $H_{AR\text{-}e} = 0$ when e is the AR itself. When $j = 1$ (no retransmissions), $C_{i,i+1(j)\_2}(h) = 0$. This fact can be derived from (7) as well.

The accumulative loads until a successful transmission of a pair of request and reply packets on the jth trial are a sum of the two portions, i.e., $C_{i,i+1(j)}(h) = C_{i,i+1(j)\_1}(h) + C_{i,i+1(j)\_2}(h)$. Therefore, in a given procedure the mean signalling loads generated by both packets under the constraint of N_{RM} transmissions are calculated as

$$\begin{aligned}
C_{i,i+1}(h) &= \sum_{j=1}^{N_{\mathrm{RM}}} C_{i,i+1(j)}(h) \\
&= \sum_{j=1}^{N_{\mathrm{RM}}} \left[q_{i,i+1}^{j-1} \cdot (1 - q_{i,i+1}) \cdot (L_i + L_{i+1}) \cdot h \right] \\
&\quad + \sum_{j=2}^{N_{\mathrm{RM}}} \left[C_{i,i+1}^{\mathrm{fail}}(h) \cdot q_{i,i+1} \cdot \frac{1 - q_{i,i+1}^{j-1}}{1 - q_{i,i+1}} \right] \\
&= (1 - q_{i,i+1}) \cdot (L_i + L_{i+1}) \cdot h \cdot \frac{1 - q_{i,i+1}^{N_{\mathrm{RM}}}}{1 - q_{i,i+1}} \\
&\quad + \frac{C_{i,i+1}^{\mathrm{fail}}(h) \cdot q_{i,i+1}}{1 - q_{i,i+1}} \cdot \sum_{j=2}^{N_{\mathrm{RM}}} (1 - q_{i,i+1}^{j-1}) \\
&= (L_i + L_{i+1}) \cdot h \cdot (1 - q_{i,i+1}^{N_{\mathrm{RM}}}) \\
&\quad + \frac{C_{i,i+1}^{\mathrm{fail}}(h) \cdot q_{i,i+1}}{1 - q_{i,i+1}} \\
&\quad \cdot \left[N_{\mathrm{RM}} - 1 - \frac{q_{i,i+1}(1 - q_{i,i+1}^{N_{\mathrm{RM}}-1})}{1 - q_{i,i+1}} \right], \\
&\quad q_{i,i+1} < 1. \qquad (8)
\end{aligned}$$

For a single packet or a pair of request and response packets transmitted between two network entities over h wired hops, since no retransmissions are invoked the signalling loads respectively are given by

$$C_i(h) = L_i \cdot h; \qquad (9)$$

$$C_{i,i+1}(h) = (L_i + L_{i+1}) \cdot h. \qquad (10)$$

In fact, by examining the first and the second terms of the right hand in (8) we can obtain $C_{i,i+1}(h) \to (L_i + L_{i+1}) \cdot h$ when $q_{i,i+1} \to 0$.

Now we can derive the expressions for the signalling loads generated in the procedures in Scheme I and Scheme II, respectively. For the user-triggered flow handoffs using Scheme I, based on Fig. 4 the mean loads in Scenarios A and B are respectively given by

$$\begin{aligned}
C_I^{\text{Usr-HO-A}} &= C_{\mathrm{RS,RA}}(1) \cdot N^{\text{HO-IF}} + C_{\text{I:BU,BA}}^{\text{Usr-HO-A1}}(H_{\text{MH-HA}}) \\
&\quad \cdot p_{\text{mv-npu}} + C_{\text{I:BU,BA}}^{\text{Usr-HO-A2}}(H_{\text{MH-HA}}) \\
&\quad \cdot (1 - p_{\text{mv-npu}}); \qquad (11)
\end{aligned}$$

$$C_I^{\text{Usr-HO-B}} = C_{\text{I:BU,BA}}^{\text{Usr-HO-B}}(H_{\text{MH-HA}}). \qquad (12)$$

The mean subtotal loads generated by both cases are thus given by

$$\begin{aligned}
C_I^{\text{Usr-HO}} &= \lambda^{\text{Usr-HO}} \\
&\quad \cdot \left[p_{\text{mv}} \cdot C_I^{\text{Usr-HO-A}} + (1 - p_{\text{mv}}) \cdot C_I^{\text{Usr-HO-B}} \right]. \qquad (13)
\end{aligned}$$

Note that the mean subnet resident time of a MN is $\left(\lambda^{\text{Usr-HO}} \cdot p_{\text{mv}}\right)^{-1}$.

For the same procedure in Scheme II, the signalling is different and so are the loads expressions. In Scenario A, the BU and BA messages are replaced with the corresponding SOAP messages; an additional pair of BU and BA is always used for the new CoA registration. In Scenario B, SOAP messages replace the MIPv6 counterparts. Therefore, in Scheme II the average subtotal loads generated by both cases are given by

$$\begin{aligned}
&C_{\text{II}}^{\text{Usr-HO}} \\
&\quad = \lambda^{\text{Usr-HO}} \cdot \left[p_{\text{mv}} \cdot C_{\text{II}}^{\text{Usr-HO-A}} + (1 - p_{\text{mv}}) \cdot C_{\text{II}}^{\text{Usr-HO-B}} \right] \\
&\quad = \lambda^{\text{Usr-HO}} \cdot p_{\text{mv}} \cdot \Big(C_{\text{II:RS,RA}}^{\text{Usr-HO-A}}(1) \cdot N_{\text{HO-IF}} \\
&\qquad + C_{\text{II:BU,BA}}^{\text{Usr-HO-A1}}(H_{\text{MH-HA}}) + (1 - p_{\text{mv-npu}}) \\
&\qquad \cdot C_{\text{II:SOAP_Req,SOAP_Rep}}^{\text{Usr-HO-A2}}(H_{\text{MH-HA}}) \Big) \\
&\qquad + \lambda^{\text{Usr-HO}} \cdot (1 - p_{\text{mv}}) \\
&\qquad \cdot \Big(C_{\text{II:SOAP_Req,SOAP_Rep}}^{\text{Usr-HO-B}}(H_{\text{MH-HA}}) \Big). \qquad (14)
\end{aligned}$$

For network-triggered flow handoffs using Scheme I, based on Fig. 5 the average subtotal loads are given by

$$\begin{aligned}
C_I^{\text{Net-HO}} &= \lambda^{\text{Net-HO}} \cdot \big(C_{\text{ICMP_Probe}}(H_{\text{AR-NIS}}) \cdot N_{\text{probe}} \\
&\quad + C_{\text{ICMP_Trigger,ICMP_Ack}}(H_{\text{NIS-HA}}) \big) / N_{\text{MH-tr}} \\
&\quad + \lambda^{\text{Net-HO}} \cdot \Big(C_{\text{BRR,BU}}^{\text{Net-HO}}(H_{\text{MH-HA}}) \Big), \qquad (15)
\end{aligned}$$

where $N_{\text{probe}} = \frac{\lambda^{\text{Probe}}}{\lambda^{\text{Net-HO}}} \cdot N_{\text{AR}}$. Note that the average loads per involved MN generated by a Trigger and a Trigger Ack are obtained by dividing the loads by $N_{\text{MH-tr}}$. This also applies to the Probe messages.

For the same procedure in Scheme II, the average subtotal loads are given by

$$C_{\mathrm{II}}^{\text{Net-HO}} = \lambda^{\text{Net-HO}} \cdot \left(C_{\text{SOAP\_Probe}}(H_{\text{AR-NIS}}) \cdot N_{\text{probe}} + C_{\text{SOAP\_Trigger,SOAP\_Ack}}(H_{\text{NIS-HA}})\right)/N_{\text{MH-tr}} + \lambda^{\text{Net-HO}} \cdot \left(C_{\text{SOAP\_Req,SOAP\_Rep}}^{\text{Net-HO}}(H_{\text{MH-HA}})\right). \quad (16)$$

Supporting procedures include initial multiple CoA registration, default flow binding policy and user profile registration, periodic flow binding policy refresh, and deregistration. Signalling loads for these procedures can be derived in a similar manner based on Figs. 2, 3, 6 and 7, respectively. For brevity, these formulae are not shown in the paper. In addition, for an oversized message whose length exceeds MTU_{min}, we assume that the message is broken into a few packets, each of which is acknowledged by the receiver. Thus, the above analysis is still applicable. This effect is taken into account when we calculate the loads to obtain the numerical results.

5.3. Message length estimation

The length of a message would directly affect the signalling loads the message generates. In this subsection, we estimate the IP-level lengths of the involved messages. Firstly, we identify the lengths of the MIPv6-based mobility messages and ICMPv6 messages based on their message formats defined in the MIPv6, ICMPv6 specifications, and the extensions we have proposed. Both MIPv6 and ICMPv6 run over UDP by default and use IPsec Encapsulating Security Payload (ESP) transport mode [37] for secure signalling. We assume the SEED-CBC algorithm [38] for encryption, and the HMAC-MD5-96 algorithm [39] for authentication. Fig. 8 illustrates the packet structure of a BU message.

In a BA or a BRR, the HoA of the MN is stored in a 24-byte Type-2 routing header instead of the Destination Options extension header of the same size used in a BU, and the BU mobility header is replaced with the corresponding BA or BRR one. In terms of length expression, these differences do not matter. Therefore, based on Fig. 8 the length of a mobility message consisting of n BIDs in procedure j is given by

$$L_{\text{MM}}^{j}(n) = 100 + \left\lceil \frac{10 + L_{\text{MM-mhdr}}^{j}(n)}{16} \right\rceil \times 16, \quad (17)$$

where $L_{\text{MM-mhdr}}^{j}(n)$ is the length of the mobility header of a BU, a BA, or a BRR. The BU case is shaded in Fig. 8. In addition, the length of an ICMPv6 message consisting of n AR prefixes (or other parameters depending on procedures) in procedure j is given by

$$L_{\text{ICMPv6}}^{j}(n) = 76 + \left\lceil \frac{14 + L_{\text{ICMPv6-payload}}^{j}(n)}{16} \right\rceil \times 16, \quad (18)$$

where $L_{\text{ICMPv6-payload}}^{j}(n)$ is the total length of the ICMPv6's own payload, which accommodates the functionality-specific customised message. Note that the lengths of the BU, BA and BRR mobility headers and the ICMPv6 payload vary from procedures or the number of BIDs, policies, AR prefixes, triggers and so forth.

Table 1 summaries the lengths of the mobility headers and the payloads of the ICMPv6 messages. All the procedures or functions refer to Scheme I unless stated otherwise. Regarding Table 1, in a given procedure or function n is the number of involved BIDs (i.e., CoA-BID bindings or interfaces) in a mobility message or the number of AR prefixes (i.e., ARs or access networks) in an ICMPv6

<table>
<tr><td>IPv6 header [CoA, HA](40 bytes)</td><td>Destination Options header [HoA](24 bytes)</td><td>ESP header (8 bytes) + IV</td><td>BU Mobility header (variable, multiple of 8 bytes)</td><td>UDP header (8 bytes)</td><td>ESP Trailer (padding, the 2-byte headers)</td><td>ESP Authentication Data</td></tr>
<tr><td colspan="3" rowspan="2">72 bytes + L(IV) (16 bytes using SEED-CBC)</td><td colspan="2">8 + $L_{BU-mhdr}^{j}(n)$ bytes (variable)</td><td>Variable (>=2 bytes)</td><td rowspan="2">Variable (12 bytes using HMAC-MD5-96)</td></tr>
<tr><td colspan="3">ESP payload (a multiple of 16 bytes if using SEED-CBC; the length of the ciphertext remain unchanged)</td></tr>
</table>

Fig. 8. Packet structure of a MIPv6 BU message.

Table 1
Lengths (bytes) of mobility headers and ICMPv6 payloads

Message	Procedure or function	Length of mobility header or ICMPv6 payload
BRR (HA → MN)	Network-triggered flow handoff	$8+8\cdot n+\left\lceil \frac{1}{8}\cdot\left(4+\sum_{i=1}^{n}\sum_{j=1}^{N_i}L_{\text{FID\_}i,j}\right)\right\rceil\cdot 8$
BU (MN → HA)	Multiple CoA registration (Schemes I, II)	$\begin{cases}12+28\cdot n & n \geqslant 1,\text{odd}\\ 16+28\cdot n & n \geqslant 2,\text{even}\end{cases}$
	Initial default policy registration	$32+8\cdot n+\left\lceil \frac{1}{8}\cdot\sum_{i=1}^{n}\sum_{j=1}^{N_i}L_{\text{FID\_}i,j}\right\rceil\cdot 8$
	User-triggered flow handoff Case A1: new CoA registration only (Schemes I, II)	$\begin{cases}12+28\cdot n & n \geqslant 1,\text{odd}\\ 16+28\cdot n & n \geqslant 2,\text{even}\end{cases}$
	User-triggered flow handoff Case A2: new CoA registration and policy update	$8+24\cdot n+\left\lceil \frac{1}{8}\cdot\left(4+4\cdot n+\sum_{i=1}^{n}\sum_{j=1}^{N_i}L_{\text{FID\_}i,j}\right)\right\rceil\cdot 8$
	User-triggered flow handoff Case B: policy update only	$32+8\cdot n+\left\lceil \frac{1}{8}\cdot\sum_{i=1}^{n}\sum_{j=1}^{N_i}L_{\text{FID\_}i,j}\right\rceil\cdot 8$
	Network-triggered flow handoff	$32+8\cdot n+\left\lceil \frac{1}{8}\cdot\sum_{i=1}^{n}\sum_{j=1}^{N_i}L_{\text{FID\_}i,j}\right\rceil\cdot 8$
	CoA and policy refresh	$8+24\cdot n+8\cdot\sum_{i=1}^{n}N_i+\left\lceil \frac{1}{8}\cdot(4+4\cdot n)\right\rceil\cdot 8$
	CoA refresh only (Scheme II)	$\begin{cases}12+28\cdot n & n \geqslant 1,\text{odd}\\ 16+28\cdot n & n \geqslant 2,\text{even}\end{cases}$
	Deregistration at the HA (Schemes I, II)	$\begin{cases}12+28\cdot n & n \geqslant 1,\text{odd}\\ 16+28\cdot n & n \geqslant 2,\text{even}\end{cases}$
BA (HA → MN)	Multiple CoA registration (Schemes I, II)	$16+8\cdot n$
	Initial default policy registration	$16+8\cdot\sum_{i=1}^{n}N_i$
	User-triggered flow handoff Case A1: new CoA registration only (Schemes I, II)	$16+8\cdot n$
	User-triggered flow handoff Case A2: new CoA registration and policy update	$16+8\cdot n+8\cdot\sum_{i=1}^{n}N_i$
	User-triggered flow handoff Case B: policy update only	$16+8\cdot\sum_{i=1}^{n}N_i$
	CoA and policy refresh	$16+8\cdot n+8\cdot\sum_{i=1}^{n}N_i$
	CoA refresh only (Scheme II)	$16+8\cdot n$
	Deregistration at the HA (Schemes I, II)	$16+8\cdot n$
ICMPv6	Trigger (NSA → HA)	$4+16\cdot N_{\text{MH-tr}}+20\cdot n+\sum_{i=1}^{n}\sum_{j=1}^{N_i}L_{\text{TID\_}i,j}$
	Trigger Ack (HA → NSA)	$4+16\cdot N_{\text{MH-tr}}+8\cdot\sum_{i=1}^{n}N_i$
	Profile (MN → NSA)	40
	Profile Ack (NSA → MN)	17
	Deregistration (MN → NSA)	17
	Deregistration Ack (NSA → MN)	17
	Probe (AR → NSA)	16

message. N_i is the number of policies or triggers bound to BID_i or AR prefix i. $L_{\text{FID\_}i,j}$ and $L_{\text{TID\_}i,j}$ are the length of FID_j option and TID_j option for BID_i and AR prefix j, respectively. In addition, based on [28] the length of a RS and its corresponding RA are 54 bytes and 104 bytes, respectively.

Next, we estimate the IP-level length of the SOAP messages. We assume SOAP over UDP for efficient signalling and fair comparison between Schemes I and II. We also assume textual-based SOAP messages using IPsec ESP and the same encryption and authentication algorithms as in Scheme I. Fig. 9 shows the structure of an IPv6 SOAP message.

Based on Fig. 9, the length of a UDP payload SOAP consisting of n BIDs in procedure j is given by

$$L^{j}_{\text{SOAP}}(n)=76+\left\lceil\frac{10+L^{j}_{\text{SOAP-hdr-payload}}(n)}{16}\right\rceil\times 16, \tag{19}$$

where $L^{j}_{\text{SOAP-hdr-payload}}(n)$ is the total length of the SOAP's own header and payload, which is shaded in Fig. 9 and varies from procedures or the number of policies, triggers or other parameters. Since SOAP is an application-level protocol, its length can hardly be precisely identified according to the message definition only. We estimate $L^{j}_{\text{SOAP-hdr-payload}}(n)$ based on empirical values regarding the length of a void (skeleton) SOAP-over-UDP message etc. from the literature [36] and express it as

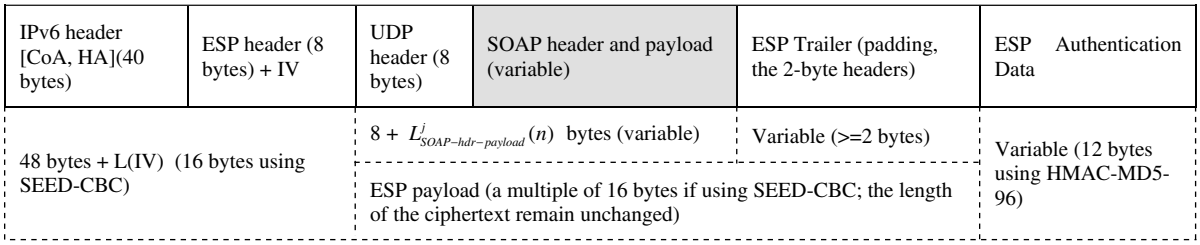


Fig. 9. Packet structure of an IPv6 SOAP message.

$$\begin{aligned} L^{j}_{\text{SOAP-hdr-payload}}(n) &= (L_{\text{void-SOAP-UDP}} - L_{\text{UDP-hdr}}) \\ &\quad + L^{j}_{\text{SOAP-hdr-extra}} + L^{j}_{\text{SOAP-body}} \\ &\cong 450 + L^{j}_{\text{SOAP-body}} \\ &\cong 450 + \sum_{i=1}^{k} L^{j}_{\text{SOAP-body-}i}, \end{aligned} \tag{20}$$

where $L_{\text{void-SOAP-UDP}}$ is the length of a void SOAP message over UDP (including the length of the UDP header, $L_{\text{UDP-hdr}}$), $L^{j}_{\text{SOAP-body}}$ is the length of the body part of a SOAP message in procedure j, $L^{j}_{\text{SOAP-hdr-extra}}$ is the length difference introduced by the header part of a SOAP message in procedure j compared with that of a void SOAP message, and $L^{j}_{\text{SOAP-body-}i}$ is the length of an item in a request such as a policy, a trigger, a profile item, a measurement in a Probe or an item in a reply to that in the request depending on whether the SOAP message is a request or a reply in procedure j. $L^{j}_{\text{SOAP-body}}$ is estimated as the accumulative length of the k policies, triggers, or other parameters enclosed in the body part of a SOAP message.

6. Numerical results

In this section, we illustrate the numerical results based on the above analyses and present corresponding explanations and discussions. Table 2 tabulates the typical values (unless stated otherwise) for the input parameters.

Firstly, Fig. 10 depicts the unit signalling loads of each procedure generated every time the procedure is invoked. In both schemes, the loads from the initial multiple CoA registration and the deregistration procedures are among the lowest and those from the default policy and profile registration are the second highest. Despite such a difference in the unit loads, their overall contribution to the total signalling loads would be unimportant owing to the very low rates at which they are provoked. The main contributors would be the remaining three procedures, whose unit loads are not low especially in Scheme II and arrival rates are much higher. Among the three procedures (and actually all the procedures), the unit loads from network-triggered flow handoffs

Table 2
Typical values for the input parameters

Parameter	Typical value
$\lambda^{\text{Init-MCoA}}$, $\lambda^{\text{Def-policy}}$, $\lambda^{\text{Usr-HO}}$, $\lambda^{\text{Net-HO}}$, $\lambda^{\text{Policy-Ref}}$, λ^{Dereg}, λ^{Probe}	0.1, 0.08, 2, 2, 2, 0.1, 60 (h^{-1})
n, N_i, $N_{\text{HO-IF}}$, $N_{\text{MH-tr}}$	2, 2, 1, 10
N_{RM}	3 for a RS [28]; 7 for the other request messages [4]
β_i	10^{-5}
p_{mv}, $p_{\text{mv-npu}}$	0.5, 0.5
$L_{\text{FID\_}i,j}$, $L_{\text{TID\_}i,j}$	36, 16 (bytes)
$L^{j}_{\text{SOAP-body-}i}$	a policy: 100, a trigger: 80, a profile: 15, a measurement in a Probe: 20, a policy refresh request item: 20, a MN's identifier used in a network trigger: 30, a deregistration request item: 15, a reply item of any request: 15 (bytes) (partially based on [40,41])
k	The number of policies or triggers or the corresponding reply items in a message: 4 (i.e., $k = n \cdot N_i$), the number of profile items: 30, the number of profile reply items: 3, the number of measurements in a Probe: 4, the number of policy refresh request items: 4 (i.e., $k = n \cdot N_i$), the number o f MNs' identifiers in a network trigger: 10 (i.e., $k = N_{\text{MH-tr}}$), the number of deregistration request or reply items: 3
$H_{\text{MH-HA}}$, $H_{\text{NIS-HA}}$, $H_{\text{AR-NIS}}$, $H_{\text{MH-NIS}}$	20, 15, 3, 4 (i.e., $H_{\text{MH-NIS}} = H_{\text{AR-NIS}} + 1$)
MTU_{min}	1280 (bytes) [42]

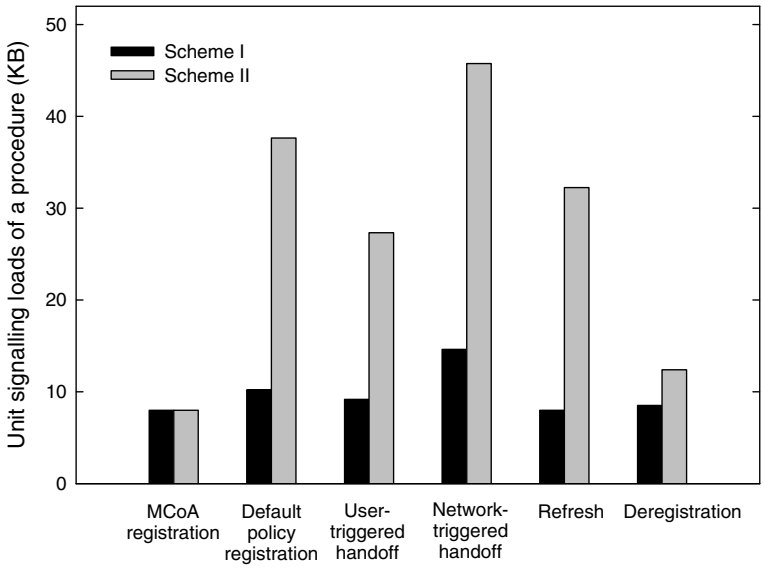


Fig. 10. Unit signalling loads of a procedure.

are the highest because of the periodic Probe messages and the triggers. These additional loads are the price paid for the network intelligence involved compared with the user-triggered handoffs. In addition, the loads from the periodic policy and multiple CoA registration refresh are by no means trivial. As to a comparison of the two schemes, the unit loads in Scheme I are extensively smaller than the corresponding ones in Scheme II in each procedure except the initial multiple CoA registration shared by both schemes. The reductions in Scheme II range from 31% to 75%. Therefore, we can predict that the total loads generated from Scheme I would be consistently lower than those from Scheme II. In the following, we scrutinise the total signalling loads from both schemes under different system conditions and quantify the differences between the two schemes.

We start with the effect of the wireless hop's BER (10^{-7}–10^{-4}) on the total loads, as shown in Fig. 11. Overall, the loads in both schemes increase with the increase of the BER. Nevertheless, in the range of 10^{-7}–10^{-5} the loads in both schemes remain almost unchanged and the gradual increase is hardly noticeable. In contrast, when the BER is larger than 10^{-5} the loads in both schemes begin to continuously rise significantly especially in Scheme II, where the remarkable increase is 191% from 10^{-5} to 10^{-4} compared with the 37% increase in Scheme I. This indicates that deteriorate channel conditions of the wireless access networks affect Scheme II in a more severe way, or Scheme I is more resilient to the error-prone wireless links. The reason lies in the fact that the lengths of the involved SOAP messages in Scheme II are much larger than their counterparts in Scheme I are; and thus the packet loss rate is much higher, which leads to more retransmissions and yields more signalling loads. In contrast to Scheme II, Scheme I can decrease the loads by 69% under typical wireless channel conditions and up to 86% under seriously lossy conditions. Note that 10^{-5} is the typical par value in Wi–Fi networks and should be satisfied normally. Therefore, despite the differences the loads provoked in Scheme II are still in an acceptable range under typical conditions.

In Fig. 12, we assume that the rate of non-movement-based user handoffs remains the default value ($1.0\ h^{-1}$) when the rate of the movement-based handoffs decreases with the increase of the subnet resident time of the MN. The total signalling loads in both schemes drop sharply first and then slowly when the mobile user becomes more and more static. The resident time 0.5 h is the turning point. The loads decrease 49% and 44% from 0.1 h to 0.5 h in Scheme I and Scheme II, respectively whilst just 18% and 15% from 0.5 h to 2.0 h. When the resident time is less than 0.5 h, the rate of movement-based handoffs is larger than $2\ h^{-1}$ and the correspondent contributed loads account for a large portion of the total loads and thus the resultant change in the total loads are more obvious. When the resident time is greater than 0.5 h, the loads contributed

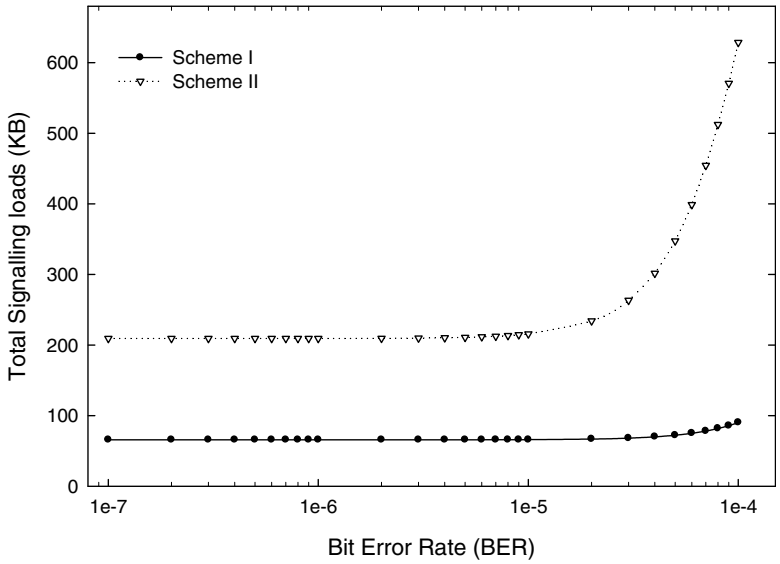


Fig. 11. Total signalling loads versus BER.

by the movement-based handoffs become more and more negligible and thus the total loads tend to be unaffected. Regarding the differences between the two schemes, Scheme I reduces 65–70% loads with the increase of a MN's subnet resident time.

Furthermore, as indicated in Fig. 13 the number of policies or triggers carried in a message grows and so do the total signalling loads when more interfaces are equipped and activated and more corresponding ARs are involved. Note that the growth of the loads is not linear in Scheme II. When the number of a MN's active interfaces is two (the default value), every messages in both schemes is not oversized and thus is carried in a single packet. When more active interfaces are used, some SOAP messages become oversized: a SOAP message signalling policies (for registration or update) would need one extra packet when the number of active interfaces reaches four and over; a SOAP trigger message would require two or three packets in case

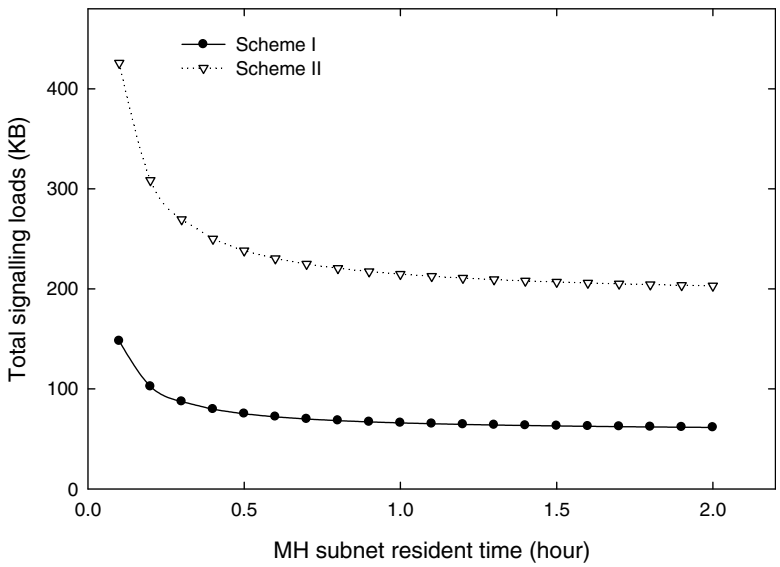


Fig. 12. Total signalling loads versus MN subnet resident time.

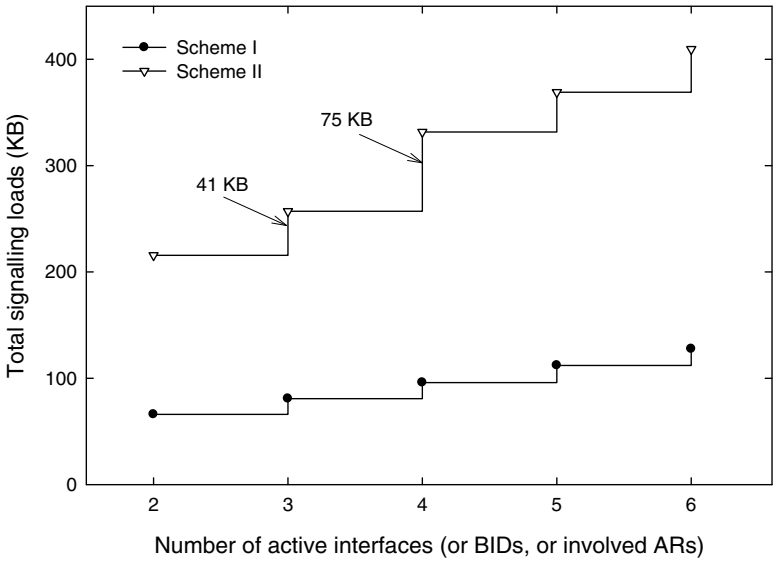


Fig. 13. Total signalling loads versus active interfaces.

of three to five interfaces and more than six interfaces, respectively. In contrast, no mobility message or ICMPv6 messages demand more than one packet in the concerned range of interfaces. Therefore, there is a much larger leap (from 41 KB to 75 KB) when the active interfaces become four in Scheme II whilst the rise is almost consistently slow and linear in Scheme I (15–16 KB). The total loads in Scheme I are 69–71% smaller than those in Scheme II. When the polices enclosed in a message increases whilst the number of active interfaces remains unchanged, the results are similar with those shown in Fig. 13 and thus they are not illustrated here for brevity.

Finally, we can sum up the above analyses. The signalling loads generated in Scheme I are consistently low mainly because that the messages based on binary MIPv6 and ICMPv6/UDP codes are much more compact. In addition, the signalling procedures in Scheme I are more efficient due to the integration of the binding refresh and the policy refresh, which are separate routines in Scheme II. In contrast, the loads invoked in Scheme II are significantly higher mainly owing to the lengthy textual SOAP messages that support MIPv6 although the loads in Scheme II are still reasonable under typical system conditions. There are a couple of potential ways to alleviate the verbosity of textual SOAP mes-

Table 3
Comparison of Scheme I and Scheme II

Signalling performance		Scheme I	Scheme II
Standardisation	Core protocol specification and development	IETF draft [9,10] and MULTINET deliverables	IETF draft [9,11] and MULTINET deliverables
	Compression	Not needed	Not standardised
	Binary characterisation	Already coded in binary format	Work in progress [44]
Interoperability	Protocol availability	MIPv6, ICMPv6 (or UDP directly)	MIPv6, SOAP (UDP or HTTP web service)
	Protocol layer	Pure network layer	Network layer and application layer
Reliability	Lost packet recovery	Retransmission (UDP)	Retransmission (UDP or TCP)
Efficiency	Signalling loads	Low (consistently)	Moderate when over UDP (can be high e.g., in highly lossy networks, or many active interfaces or policies)
Security	Encryption, authentication	IPsec	IPsec or HTTPS
Extensibility	Modifiability to developers	Moderate	High (regarding SOAP)

sages. Firstly, it is noted that at the costs of extra processing requirements and overheads, compression would be helpful to shrink the size of very large SOAP messages. Unfortunately, it is well observed [40,43] that for SOAP messages smaller than the minimum MTU (1280 bytes [42]) compression may actually expand the length of the message due to the intrinsic overheads in a compression algorithm such as gzip. Alternatively, binary encoding or characterisation is another prospective approach. Nevertheless, this approach would raise interoperability concerns since no standard SOAP binary encoding has been widely adopted though this work is in progress in W3C [44] towards a standard solution. Table 3 lists a comparison of Scheme I and Scheme II based on the proposed design, the analytical results, and other considerations. We may conclude that the two proposed schemes would adequately satisfy most of the common design requirements in typical scenarios. Especially, Scheme I boasts its high signalling efficiency whilst Scheme II is more advantageous in message extensibility thanks to the XML-based formats of the SOAP messages.

7. Implementations and experimental results

To complement the analytical work and further assess the performance of the proposed signalling schemes, we have constructed a local IPv6 wireless testbed. In this section, we present our testbed implementations and experimental results.

7.1. Testbed setup

Note that in principle our proposed schemes are applicable to both MIPv6 and its network mobility extension NEMO. We chose NEMO for demonstration in our testbed since the available NEMO implementation NEPL [45] is more mature in multihoming support compared with its MIPv6 counterpart MIPL [46].

Table 4 lists the hardware and software settings. A set of Linux PCs are configured to act as the NSA, the CN, the HA and the NEMO mobile router (MR) with two Wi-Fi cards. The mobile network node (MNN) is a Windows XP PC in the mobile network whose multihoming and mobility proxy is the MR. One can deem that the MR plus the MNN is equivalent to the mobile host in MIPv6. The NSA, the network trigger generator, is collocated with the CN, which is a video streaming server and a FTP server for the MNN. The VLC [47] and the proftpd [48] software are used for video streaming and FTP, representing typical real-time and non-real-time applications respectively. A couple of 802.11 b/g ARs with OpenWRT [49] as operating system provide the multihomed MR with two wireless connections (and thus two separate routes between the HA and the MR). The bit rate of the wireless channels is set to be 11 Mbps by default during the following experiments. The testbed topology is illustrated in Fig. 14.

The handoff execution functionality was built upon the NEMO implementation NEPL with integrated MCoA support. Once a trigger is received and parsed, the enclosed policies are installed and enforced with the ip6tables at the HA and the MR. The subsequent packets meeting a policy are marked with the corresponding BID, e.g., 100 or 200. A routing table per BID is generated, e.g., routing tables # 100 and # 200. These tables are looked up for forwarding the marked packets to the corresponding interfaces. IPv6 stateless host auto-configuration was achieved through the radvd module [50]. The multihomed MR automatically configures a CoA for each of its interfaces and registers the CoAs with the HA via the MCoA support. Based on the proposed Schemes I and II, we have implemented two

Table 4
Testbed hardware and software

Node	Hardware	Third-party software	Software
NSA, CN	VIA Samuel 2 (600 MHz, 512 MB RAM)	Apache2, PHP5, proftpd (FTP server), VLC (video server)	Scheme I: UDP_NSA Scheme II: SOAP_NSA
HA	VIA Nehemiah (1.00 GHz, 256 MB RAM)	Apache2, PHP5, NEPL + MCoA, radvd	Scheme I: UDP_HA Scheme II: SOAP_HA
MR	VIA Nehemiah (1.20 GHz, 512 MB RAM)	Apache2, PHP5, NEPL + MCoA, radvd	Scheme I: UDP_MR Scheme II: SOAP_MR
MNN	Pentium D (2.80 GHz, 1 GB RAM)	ftp (FTP client), VLC (video client)	
AR1, AR2	Linksys WRT54GL	OpenWRT, radvd	

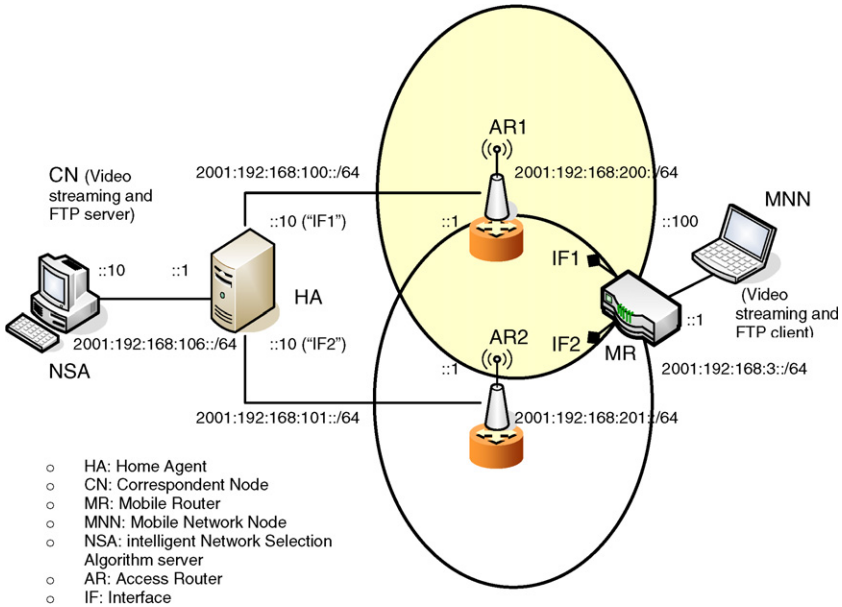


Fig. 14. Testbed topology.

signalling methods. The first is a variant of Scheme I: it operates over UDP (the ICMPv6 header is optional) whereas it does not modify the mobility messages directly as the MIPv6 BRR message has not been available in the current NEMO. The other one is a HTTP-version realisation of Scheme II. It was coded in SOAP messages and developed with PHP5 [51], which has built-in C-based SOAP over HTTP (SOAP over UDP has not been available in PHP5). Apache2 [52] was installed as the HTTP server. Both implemented schemes were verified through experiments to achieve the distribution and enforcement of dynamic triggers. Each trigger is composed of one or more policies, which are enforced by manipulating the ip6tables.

7.2. Experimental results

On our testbed, we have designed and conducted a set of experiments including validation and comparisons of the two implemented signalling schemes and measuring handoff signalling delays, validation of handoffs of different application flows between interfaces, subjective perception of a real-time application (video streaming), and objective evaluation of a non-real-time application (FTP file downloading). In the experiments, we focused on network-initiated flow handoffs triggered by intelligent network selection rather than user movement due to the targeted user scenario in MULTINET.

7.2.1. Handoff signalling delay

We started with assessing the implementations of the proposed schemes with the handoff signalling delays measured. The handoff signalling delay is defined as the elapsed time between a trigger is generated by the NSA and a final acknowledgement is received at the NSA from the HA (on completing the distribution and enforcement of both the trigger at the HA and the symmetric trigger at the MR). Fig. 15 depicts the handoff signalling delay in both implementations when the number of policies coded in a trigger varies. Each policy consists of a full five-tuple to identify a flow and a BID [9] to identify an interface. The action of each policy is adding. Experiments for each scenario were performed repeatedly and the mean values are reported here. We have examined three cases, where the two mechanisms operated in the default 11-Mbps wireless channel to compare their performances and additionally the UDP-based mechanism (Scheme I) operated under the 1-Mbps condition to show the effects of the wireless channel bandwidth.

Firstly, as expected the delays in all the cases increase with the growth of the number of policies because of the increasing latencies invoked for transmitting and processing more policies. Despite all the differences to be discussed, all the delays are well under one second. Furthermore, the delays generated under conditions of the lower date rate (1 Mbps) are higher than those with the higher date

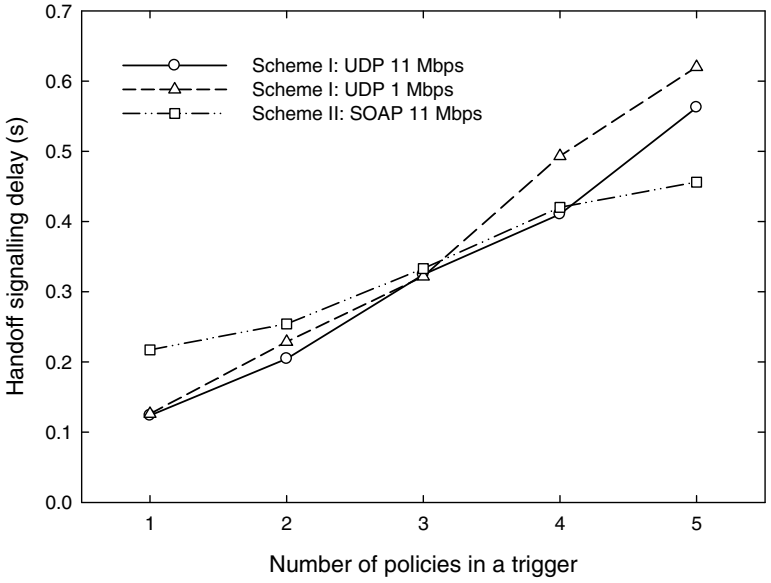


Fig. 15. Handoff signalling delay vs. number of policies in a trigger.

rate (11 Mbps) are as shown in the "UDP 1 M" and "UDP 11 M" cases. The relative difference between the signalling delays achieved by each scheme increases with the number of policies since each more policy tends to enlarge the difference.

Secondly, overall the two schemes appear to incur comparable delays when both operated in 11 Mbps in our local testbed. When the number of policies in a trigger is small (one or two), the UDP one generates a clear-cut lower delay although the difference is decreasing with the number of policies increases. When three or four policies are coded in a trigger, the difference in the delays is negligible. In the scenario of five policies in a trigger, the SOAP approach yields a significantly lower delay. Therefore, it seems that the UDP scheme is more effective when the number of policies in a trigger is small whilst the SOAP one seems more advantageous when the number becomes high. The reason behind this observation seems to be implementation specific. The SOAP-based implementation utilised the built-in XML object handling for parsing and processing the policies, and thus Scheme II appears more efficient for larger number of policies.

It is noted that the handoff signalling delay in the SOAP/HTTP-based mechanism includes an additional latency for the three-way TCP handshake for each connection establishment between the nodes, whilst the delay excludes the latency for fetching the web services description language

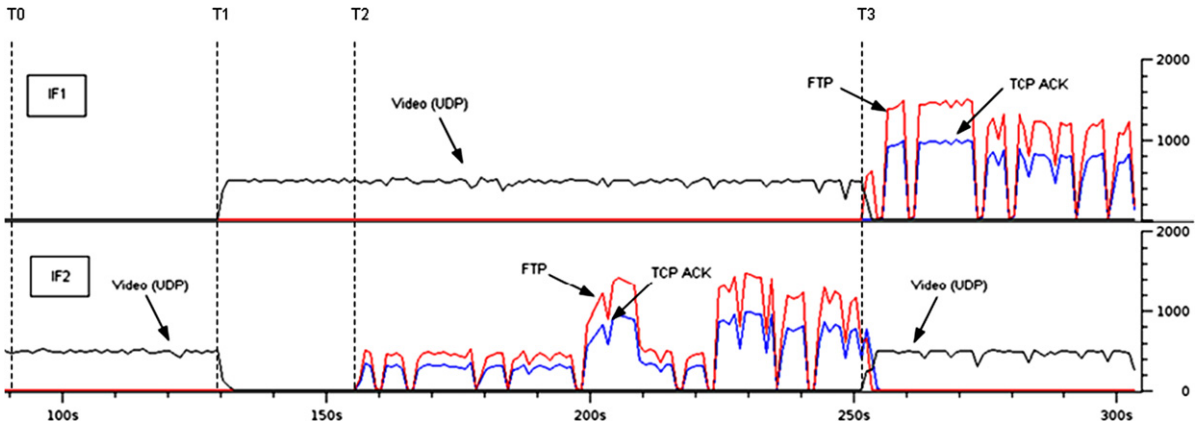


Fig. 16. Policy-based flow handoffs between interfaces.

(WSDL) files. The WSDL files define the proposed network services. The latency for fetching the WSDL files is only incurred at the first time before a trigger is ever transferred or when the local WSDL caches expire after 24 h. In contrast, no connection delays (or the WSDL latency) are incurred in the UDP-based approach. Nevertheless, in our localised high-speed testbed, the TCP connection delays are insignificant since the transmissions of messages are overwhelmingly fast.

We conjecture that when the schemes are deployed in a wide-area network (WAN), the latency between the HA and the AR would lead to a disadvantage for the SOAP/HTTP mechanism as three additional one-way Internet delays would be incurred for connection setup. It is expected that the UDP approach would avoid such extra delays. Consequently, the flow handoff process would be accelerated so that the aim of the handoff (typically for load sharing/balancing to the targeted nomadic users) can be achieved sooner. We also note that signalling delay would rise with the WAN propagation delay. Nevertheless, typically such additional signalling delay would not effectively affect the smoothness of flow handoffs. The reason is as follows. In wireless overlays networks, multiple interfaces connect to their serving ARs simultaneously. During a flow handoff, the original interface is being used continuously until the flow is handed over to the target interface. This is the advantage of multihoming-based handoffs in contrast to traditional break-before-make handoffs. Therefore, it is envisioned that smooth flow handoffs would still be achievable even in a WAN environment. In future work, we would enhance our current testbed for further investigations in broader scenarios. Part of the enhancements would include a WAN emulator such as netem or NIST Net.

7.2.2. Flow handoff effectiveness

Next, we have performed numerous tests to verify the effectiveness of the proposed policy-based flow handoffs. In the following, we present a case study of flow handoffs of two different applications, video streaming and FTP file downloading. As illustrated in Fig. 16, the X axis represents the time sequence of the experiment whilst the Y axis shows the traffic volume of the application flows. Table 5 lists the details of the involved policies. A flow in this case is identified by a four-tuple: source address (srcAddr), destination address (dstAddr), source or

Table 5
Triggers and policies

Time instance	Trigger	Policy						Symmetric policy					
		srcAddr	dstAddr	srcPort	dstPort	Protocol	IF	srcAddr	dstAddr	srcPort	dstPort	Protocol	IF
T1	Trigger 1	2001:192:168:106::10	2001:192:168:3::100		1234	UDP	IF1						
T3	Trigger 2	2001:192:168:106::10	2001:192:168:3::100	20		TCP	IF1	2001:192:168:3::100	2001:192:168:106::10		20	TCP	IF1
		2001:192:168:106::10	2001:192:168:3::100		1234	UDP	IF2						

destination port number (srcPort or dstPort), and transport protocol.

At T0, a RTP/UDP-based video streaming has been established and is being transmitted from the CN towards the MNN via the MR. By default, all traffic travels through IF2 of the MR and the corresponding interface of the HA.

At T1, the NSA issued Trigger 1, which contains one policy to be enforced at the HA. The policy indicates that the ongoing video streaming should be handed off from IF2 to IF1. The symmetric policy for the MR is optional as the streaming is a one-way traffic (no uplink traffic for acknowledgement or reverse streaming). After the policy is enforced at the HA, the streaming flow is handed over from IF2 to IF1.

This handoff decision could be made according to the output of the intelligent network selection algorithm for load sharing, fault tolerance or user/application preferences. For instance, the NSA had detected that the route via IF1 was underutilised or the route via IF2 was overloaded (or temporarily going down). It could also result form an establishment request of a new application session (e.g., the subsequent FTP downloading), which has the priority to use the IF2 route.

At T2, a FTP file transfer was initiated by the MNN to download a large file from the CN. Again, by default the FTP flow (from the CN to the MNN) and the TCP ACK flow (from the MNN to the CN) were transmitted through IF2.

At T3, the NSA issued Trigger 2, comprising two policies. One policy is to switch the FTP flow from IF2 to IF1. A symmetric policy was also generated at the HA for the MR to redirect the TCP ACKs together. Meanwhile, the other policy demands that the video streaming be handed off from IF1 to IF2. The symmetric policy to the latter one is optional. Consequently, the three flows were handed over to the targeted interfaces, respectively.

Such experiments were repeated with random flow handoffs of the applications between the interfaces. When the flow handoffs occurred in these experiments, no noticeable disruptions were perceived by a number of observers viewing the video streaming at the MNN. Therefore, it seems that the flow handoffs do not significantly degrade the subjective QoS of real-time applications.

Further experiments were carried out to investigate the performance of the FTP/TCP applications under flow handoff conditions. Fig. 17 demonstrates a typical result showing the sequence numbers of the received TCP segments at the MNN as the time of a FTP downloading elapsed. As shown in Fig. 17, the TCP segments received at the MNN are in good order all the time despite the fact that a couple of flow handoffs were experienced at the time instances around 11 s and 34 s, respectively. It is noted that the time instances on the X axis are not evenly distributed since the FTP downloading is not of constant bit rate (CBR) as can be found from Fig. 16. Such in-sequence delivery was also found in additional flow handoff experiments for a TCP flow of CBR. As illustrated in Fig. 18 (with the X axis plotted to scale), although three handoffs occurred during the session, packets were well received.

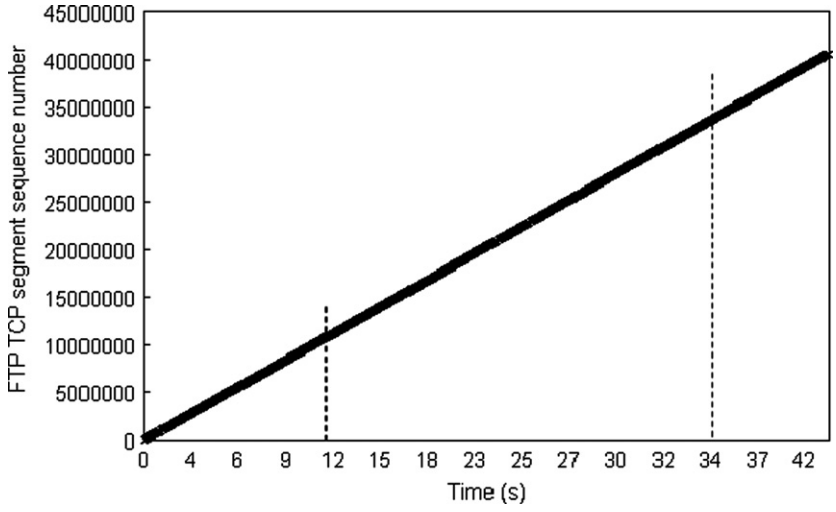


Fig. 17. FTP flow handoff performance.

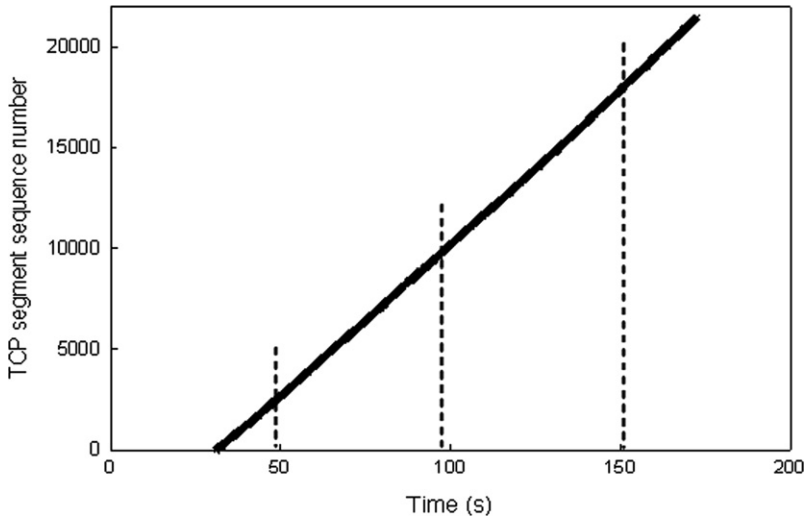


Fig. 18. TCP (CBR) flow handoff performance.

The experiments presented in this section have validated the overall design and provided preliminary performance results of each signalling scheme. The experiments illustrate that the handoff mechanisms are appropriate when both real-time and non-real-time applications are involved.

8. Conclusion

To advance the design and evaluation of flow handoff support for multihomed mobile users in wireless overlay networks, we have proposed and evaluated two signalling schemes to achieve comprehensive and standard-oriented flow handoff management. The proposed schemes greatly extend and optimise the work-in-progress specifications in the IETF to support both user- and network-triggered flow handoffs based on MIPv6/NEMO. Both handoff methods are able to meet the requirements imposed by applications and by the network provider. The major differences between the two schemes lie in signalling efficiency and message extensibility.

Scheme I utilises the potentials of mobility messages with supporting ICMPv6/UDP message defined. The design of the binary-coded message formats is not straightforward although the resultant messages would be very compact and thus would be outstandingly bandwidth efficient. The analyses on signalling loads clearly highlight this advantage under various conditions. The numerical results indicate that Scheme I can reduce around 70% total signalling loads in most cases compared with Scheme II. On the other hand, Scheme II is based on textual SOAP messages. Nevertheless, Scheme II still appears reasonable in signalling efficiency under typical conditions. Furthermore, the XML-based SOAP messages are organised in a much more readable, editable, and extensible way and thus Scheme II seems understandably more appealing to developers.

Preliminary implementations in our testbed have validated the concepts and effectiveness of the proposed designs in achieving policy-based flow handoffs. The experimental results show that smooth flow handoffs triggered by the network for nomadic users seems achievable for both real-time and non-real-time applications. In addition, when transmission delays are low as in broadband hybrid wired and wireless networks as demonstrated in our testbed, the processing delays would dominate the flow handoff signalling delays. In that context, the two schemes seem to generate comparable handoff signalling delays.

In conclusion, Scheme I outperforms Scheme II in signalling efficiency, which would be desired especially when links of narrow bandwidth are involved. In contrast, Scheme II is superior in signalling readability and extensibility, which have well facilitated the integration process in our joint project. Both schemes invoke decent flow handoff signalling delays and smooth flow handoffs seem promising. Future work would further investigate the effects of the proposed schemes in large-scale networks.

Acknowledgement

This work is sponsored by the EU IST MULTINET project (www.ist-multinet.org). The authors would like to thank for our project partners involved in the discussions on this topic. The authors also appreciate the anonymous reviewers' insightful remarks and scrupulous editing.

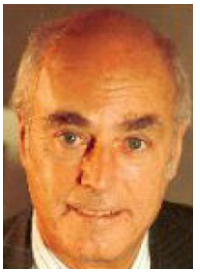
References

[1] Y.-B. Lin, A.-C. Pang, Wireless and Mobile All-IP Networks, Wiley, New York, 2005.
[2] E. Gustafsson, A. Jonsson, Always best connected, IEEE Wireless Communications 10 (1) (2003) 49–55.
[3] O. Lazaro, A. Gonzalez, L. Aginako, T. Hof, F. Sidoti, P. Vaquero, et al., MULTINET: enabler for next generation services, in: Proceedings of the 17th Wireless World Research Forum (WWRF) Meeting, Heidelberg, Germany, November 2006.
[4] D.B. Johnson, C. Perkins, J. Arkko, Mobility support in IPv6, IETF RFC 3775, June 2004.
[5] H. Soliman, C. Catelluccia, K.E. Malki, Ludovic Bellier, Hierarchical mobile IPv6 mobility management (HMIPv6), IETF RFC 4140, August 2005.
[6] R. Koodli (Ed.), Fast handovers for mobile IPv6, IETF RFC 4068, Jul 2005.
[7] V. Devarapalli, R. Wakikawa, A. Petrescu, P. Thubert, Network mobility (NEMO) basic support protocol, IETF RFC 3963, January 2005.
[8] M. Yabusaki, T. Okagawa, K. Imai, Mobility management in all-IP mobile network: end-to-end intelligence or network intelligence? IEEE Communications Magazine 43 (2) (2005), supl. 16–supl. 24.
[9] R. Wakikawa, T. Ernst, K. Nagami, Multiple care-of addresses registration, IETF Internet Draft. <draft-wakikawa-mobileip-multiplecoa-05.txt>, work in progress, February 2006.
[10] H. Soliman, N. Montavont, N. Fikouras, K. Kuladinithi, Flow bindings in mobile IPv6, IETF Internet Draft. <draft-soliman-monami6-flow-binding-02.txt>, work in progress, September 2006.
[11] K. Mitsuya, K. Tasaka, R. Wakikawa, A schema fragment for flow distribution, IETF Internet Draft. <draft-mitsuya-monami6-flow-distribution-00.txt>, work in progress, June 2006.
[12] M. Gudgin, M. Hadley, J.-J. Moreau, H.F. Nielsen, SOAP 1.2 part 1: messaging framework, W3C Recommendation, June 2003.
[13] R. Stewart, Q. Xie, K. Morneault, et al., Stream control transmission protocol, IETF RFC 2960, October 2000.
[14] J.R. Iyengar, P.D. Amer, R. Stewart, Concurrent multipath transfer using SCTP multihoming over independent end-to-end paths, IEEE/ACM Transactions on Networking 14 (2006) 951–964.
[15] S.J. Koh, M.J. Chang, M. Lee, mSCTP for soft handover in transport layer, IEEE Communications Letters 8 (3) (2004) 189–191.
[16] A. Argyriou, V.K. Madisetti, A soft-handoff transport protocol for media flows in heterogeneous mobile networks, Computer Networks 50 (2006) 1860–1871.
[17] R. Moskowitz, P. Nikander, Host identity protocol architecture, IETF RFC 4423, August 2005.
[18] IEEE P802.21™/D00.01, Draft IEEE specification for local and metropolitan area networks: media independent handover services, July 2005.
[19] N. Yamai, K. Okayama, H. Shimamoto, T. Okamoto, A dynamic traffic sharing with minimal administration on multi-homed networks, in: Proceedings of the IEEE International Conference on Communications (ICC'01), Helsinki, Finland, June 2001.
[20] D.S. Phatak, T. Goff, J. Plusquellic, IP-in-IP tunnelling to enable the simultaneous use of multiple IP interfaces for network level connection striping, Computer Networks 43 (2003) 787–804.
[21] C. Huang, C. Tsai, P. Su, MultiGate6: an IPv6 multihoming gateway using a hybrid approach, Computer Communications 29 (2006) 1842–1857.
[22] H.J. Wang, R.H. Katz, J. Giese, Policy-enabled handoffs across heterogeneous wireless networks, in: Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA'99), New Orleans, USA, February 1999.
[23] S. Balasubramaniam, J. Indulska, Vertical handover supporting pervasive computing in future wireless networks, Computer Communications 27 (2004) 708–719.
[24] D.K. Goldenberg, L.L. Qiu, H.Y. Xie, Y.R. Yang, Y. Zhang, Optimising cost and performance for multihoming, Computer Communications Review 34 (2004) 79–92.
[25] K. Shima, Y. Uo, N. Ogashiwa, S. Uda, Operational experiment of seamless handover of a mobile router using multiple care-of address registration, Journal of Networks 1 (3) (2006) 23–30.
[26] S. Kent, K. Seo, Security architecture for the Internet protocol, IETF RFC 4301, December 2005.
[27] J. Arkko, V. Devarapalli, F. Dupont, Using IPsec to protect mobile IPv6 signalling between mobile nodes and home agents, IETF RFC 3776, June 2004.
[28] T. Narten, E. Nordmark, W. Simpson, Neighbour discovery for IP version 6 (IPv6), IETF RFC 2461, December 1998.
[29] S. Thomson, T. Narten, IPv6 stateless address autoconfiguration, IETF RFC 2462, December 1998.
[30] Q. Wang, M.A. Abu-Rgheff, IPv6-based architecture for fast and cost-effective micro-mobility management, in: Proceedings of the IEE 6th International Conference on 3G and Beyond (IEE 3G2005), London, UK, November 2005.
[31] Y.-H. Han, S.-H. Hwang, Movement detection analysis in mobile IPv6, IEEE Communications Letters 10 (1) (2006) 59–61.
[32] M. Bafutto, P.J. Khn, G. Willmann, Capacity and performance analysis of signalling networks in multivendor environments, IEEE Journal on Selected Areas in Communication 12 (3) (1994) 490–500.
[33] Q. Wang, M.A. Abu-Rgheff, Signalling analysis of cost-efficient mobility support by integrating mobile IP and SIP in all IP wireless networks, International Journal of Communication Systems 19 (2) (2006) 225–247.
[34] I.F. Akyildiz, W. Wang, A dynamic location management scheme for next generation multi-tier PCS systems, IEEE Transactions on Wireless Communications 1 (1) (2002) 178–189.

[35] M. Gudgin (Ed.), SOAP-over-UDP, Technical Specification, September 2004.
[36] K.A. Phan, Z. Tari, P. Bertok, A benchmark on SOAP's transport protocols performance for mobile applications, in: Proceedings of the 2006 ACM Symposium on Applied Computing (SAC'06), Dijon, France, April 2006.
[37] S. Kent, R. Atkinson, IP encapsulating security payload (ESP), IETF RFC 2406, November 1998.
[38] H.J. Lee, J.H. Yoon, S.L. Lee, J.I. Lee, The SEED cipher algorithm and its use with IPsec, IETF RFC 4196, October 2005.
[39] C. Madson, R. Glenn, The use of HMAC-MD5-96 within ESP and AH, IETF RFC 2403, November 1998.
[40] J. Kangasharju, T. Lindholm, S. Tarkoma, Requirements and design for XML messaging in the mobile environment, in: Proceedings of the 2nd International Workshop on Next Generation Networking Middleware, Waterloo, Canada, May 2005.
[41] A. Ng, S. Chen, P. Greenfield, An evaluation of contemporary commercial SOAP implementations, in: Proceedings of the AWSA'04, Melbourne, Australia, April 2004.
[42] S. Deering, R. Hinden, Internet protocol, version 6 (IPv6) specification, IETF RFC 2460, December 1998.
[43] A. Ng, P. Greenfield, S. Chen, A study of the impact of compression and binary encoding on SOAP performance, in: Proceedings of the 6th Australasian Workshop on Software and System Architecture (AWSA'05), Brisbane, Australia, March 2005.
[44] O. Goldman, D. Lenkov (Eds.), XML binary characterisation, W3C Working Group Note, work in progress, Mar 2005.
[45] NEMO Implementation for Linux (NEPL). <http://www.nautilus6.org/nemo/>.
[46] Mobile IPv6 for Linux (MIPL). <http://mobile-ipv6.org/>.
[47] VLC media player. <http://www.videolan.org/vlc/>.
[48] Proftpd FTP server. <http://www.proftpd.org/>.
[49] OpenWRT. <http://openwrt.org/>.
[50] Linux IPv6 Router Advertisement Daemon (radvd). <http://www.litech.org/radvd/>.
[51] Hypertext Preprocessor (PHP). <http://www.php.net/>.
[52] Apache HTTP Server. <http://httpd.apache.org/>.

Qi Wang received his BEng in Electronic Engineering and MEng in Communication and Electronic System from Dalian Maritime University, China, in 1995 and 1998, respectively; and his PhD in Mobility Support Architectures for Next-Generation Wireless Networks from the University of Plymouth, UK, in 2006. He was granted a multi-year British ORS award for his PhD programme. From 1998 to 2001, he was with the State Grid Corporation of China (Shandong) as an ICT engineer. Since 2006, he has been working on an EU IST FP6 project MULTINET as a postdoctoral research fellow with the University of Strathclyde, UK. His current research interests include IP networks, mobility management and multihoming support. He is a member of IEEE and on the technical programme committees of IEEE PIMRC'08 and CCNC'09.

Robert Atkinson is a Lecturer for Communications and Signal Processing, University of Strathclyde, UK. He completed his PhD in Mobile and Wireless Communications at Strathclyde in 2003. Throughout his time at the University he has worked on a variety of topics from Medium Access Control to Heterogeneous Networking. He is actively researching intelligent access network selection, user mobility solutions and Ad Hoc Networking. He is a senior member of IEEE and member of IET.



John Dunlop received the BSc degree in electrical engineering from University College of Swansea in 1966 and the PhD degree in telecommunications from the University of Wales in 1970. He is currently a Professor of Electronic Systems Engineering and Head of the Mobile Communications Group, University of Strathclyde, UK. He has recently completed a three-year term as a Director of the UK Virtual Centre of Excellence in Mobile and Personal Communications. He has been involved in research programs in communication systems and electronic systems engineering for more than two decades. This includes participation in RACE Definition Phase, RACE Mobile Communications Project (R1043), RACE Advanced Time Division Multiple Access ATDMA (R2084), and ACTS Mobile Communications Services for High-Speed Trains MOSTRAIN (AC104) and as a full academic member of the UK Virtual Centre of Excellence in Mobile and Personal Communications. He has held several UK Engineering and Physical Sciences Research Council (EPSRC) awards on local area communications, underwater communications, and mobile communications and holds contracts from the Mobile VCE covering work in the areas of Networks and Services. He is also holder of several contracts with mobile communications companies. He is author and co-author of more than 150 scientific papers on Electronics Systems Engineering and Communications Engineering in international journals and conferences. He is co-author of *Telecommunications Engineering* which has been adopted as a standard text in many British Universities and a co-author of *Digital Mobile Communications and the TETRA System* (New York: Wiley, 1999).